Beyond Natural Language: LLMs Leveraging Alternative Formats for Enhanced Reasoning and Communication

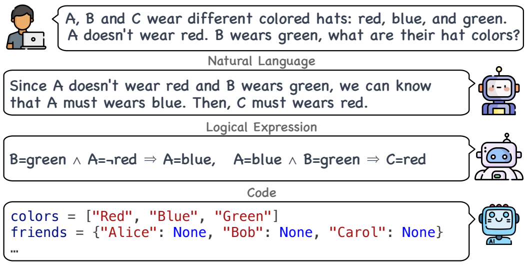
Weize Chen[1]*, Chenfei Yuan[1]*, Jiarui Yuan[1]*, Yusheng Su[1], Chen Qian[1],
Cheng Yang[3]†, Ruobing Xie[2], Zhiyuan Liu[1]†, Maosong Sun[1]
[1] Tsinghua University
[2] Tencent
[3] Beijing University of Posts and Telecommunications
{chenwz21,yuancf21,yuanjr22}@mails.tsinghua.edu.cn

Abstract

Natural language (NL) has long been the predominant format for human cognition and communication, and by extension, has been similarly pivotal in the development and application of Large Language Models (LLMs). Yet, besides NL, LLMs have seen various non-NL formats during pre-training, such as code and logical expression. NL's status as the optimal format for LLMs, particularly in single-LLM reasoning and multi-agent communication, has not been thoroughly examined. In this work, we challenge the default use of NL by exploring the utility of non-NL formats in these contexts. We show that allowing LLMs to autonomously select the most suitable format before reasoning or communicating leads to a 3.3 to 5.7% improvement in reasoning efficiency for different LLMs, and up to a 72.7% reduction in token usage in multi-agent communication, all while maintaining communicative effectiveness. Our comprehensive analysis further reveals that LLMs can devise a format from limited task instructions and that the devised format is effectively transferable across different LLMs. Intriguingly, the structured communication format decided by LLMs exhibits notable parallels with established agent communication languages, suggesting a natural evolution towards efficient, structured communication in agent communication. Our code is released at https://github.com/thunlp/AutoForm.

1 Introduction

Natural language (NL) has long been recognized as a fundamental format for human thought expression and communication, underscored by its pivotal role in the cognitive processes and information exchange of humans (Chomsky, 2006; Lakoff, 2008; Whorf, 2012). However, the human mind's capabilities often extend beyond the scope of NL, as suggested by the concept of *mentalese*, a mental language posited by linguists (Fodor, 1975; Pinker, 2003). Recent advancements in LLMs (OpenAI, 2023b; Google et al., 2023; Anthropic, 2023) have been remarkable, leading to their increasingly sophisticated application in language agents (Yao et al., 2023b; Park et al., 2023; Significant Gravitas). These advancements, while impressive, predominantly utilize NL for both single-LLM reasoning via Chain-of-Thought (CoT) (Wei et al., 2022; Kojima et al., 2022) and multi-agent communication (Wu et al., 2023; Park et al., 2023). Given the human mind's proficiency in transcending NL, critical inquiries emerge: Is NL the optimal format for LLMs in reasoning and inter-agent communication? If not, how should we determine the most suitable format for these applications (Fig. 1)?



A, B and C wear different colored hats: red, blue, and green. A doesn't wear red. B wears green, what are their hat colors?

Since A doesn't wear red and B wears green, we can know that A must wears blue. Then, C must wears red.

B=green ∧ A=¬red ⇒ A=blue,   A=blue ∧ B=green ⇒ C=red

```
colors = ["Red", "Blue", "Green"]
friends = {"Alice": None, "Bob": None, "Carol": None}
…
```

Figure 1: LLMs may leverage non-NL thought format.

Recent research challenges the notion that NL is the ideal intermediate format for LLM reasoning and multi-agent communication. Emerging variants of CoT, such as Program-of-Thought (Chen et al., 2022; Gao et al., 2023) and X-of-Thought (Liu et al., 2023) have explored the use of alternative formats like code and mathematical equations, expanding the LLMs' reasoning capabilities. However, these approaches often integrate external tools, where the alternative formats primarily serve as a means to facilitate tool execution (e.g., prompting LLM to generate code and use code interpreter execution result as the answer). This

* Equal Contribution
† Corresponding author.

Findings of the Association for Computational Linguistics: EACL 2024, pages 10626–10641
November 12-16, 2024 ©2024 Association for Computational Linguistics

introduces complexity in discerning whether the performance improvements are attributable to the format itself or the accompanying tools. Additionally, while the natural ambiguities and emotions inherent in NL may align well with the nuances of human communication, these may not be desired in agent communication, where precision is more important. Nonetheless, current multi-agent research predominantly utilizes NL (Li et al., 2023a; Wu et al., 2023; Chen et al., 2023), with limited exploration of other potentially more accurate and efficient communication formats.

In this study, we implement a straightforward and effective mechanism that prompts the model to favor non-NL formats for single-LLM reasoning and multi-agent communication tasks. By adding an instruction to the original CoT prompt that directs LLMs to explore a non-NL format appropriate for the current input, we showcase the LLMs' capacity for autonomous format decisions. We observe that the LLMs can leverage many non-NL formats such as ordered lists, logical expressions, and markdown tables to reason better. Also, we observe that agents can use more structured language as their communication language to enhance the efficiency of multi-agent collaboration. In particular, our analyses across various single-LLM reasoning tasks demonstrate an average improvement in performance by 3.3-5.7%. For multi-agent communication, we observe a reduction in token usage by up to 72.7% without sacrificing effectiveness. These results highlight the considerable potential of non-NL formats in amplifying the reasoning capabilities and communicative efficiency of LLMs.

Our investigation further extends to a comprehensive analysis revealing that LLMs can devise a suitable format from a set of task-specific examples. Using the fixed devised format for the whole task also leads to better answers. Moreover, we show that the formats devised by one LLM are transferable to another LLM. And for the multi-agent communication format, we find that the format adopted by LLMs mirrors those of traditional Agent Communication Languages (ACLs) like KQML (Finin et al., 1994), highlighting their clarity, brevity, and structured format for efficient exchanges. Empirically, our approach significantly reduces token usage compared to both ACLs and NL without sacrificing performance. Our work underscores the efficacy of non-NL formats in advancing LLM reasoning and communication.

2 Related Work

LLM Reasoning. LLMs have exhibited impressive reasoning performance, especially when employing Chain-of-Thought (CoT) technique (Wei et al., 2022; Kojima et al., 2022). CoT requires LLMs to articulate their reasoning process step-by-step before arriving at a final answer. Building on the CoT framework, variants have been proposed. Program-of-Thought (PoT) (Chen et al., 2022; Gao et al., 2023) prompts models to generate code as thought, and offloads the answer generation to a code interpreter. X-of-Thought (Liu et al., 2023) integrates CoT, PoT and Equation-of-Thought, dynamically ensembling these methods for improved reasoning. Tree-of-Thought (Yao et al., 2023a) employs depth and breadth-first search techniques to produce high-quality reasoning chains. While some CoT variants explore formats beyond NL for reasoning, the chosen formats' improvements are obscured by the concurrent use of supplementary tools such as code interpreters, blurring the distinction between format efficacy and tool execution. We focus on the format itself, investigating whether alternative formats to NL improve the CoT performance.

Multi-Agent Problem Solving. Advances in Large Language Models (LLMs) have led to the development of autonomous agents like AutoGPT (Significant Gravitas) and OpenAI Assistant (OpenAI, 2023a), demonstrating success in diverse tasks (Shinn et al., 2023; Mialon et al., 2023; Zhou et al., 2023; Boiko et al., 2023). Recent research extends this to multi-agent systems for collaborative problem-solving (Du et al., 2023; Osika, 2023; Hong et al., 2023; Qian et al., 2023). CAMEL (Li et al., 2023a) explores collaborative problem-solving between two agents through role-playing. ChatEval (Chan et al., 2023) and PRD (Li et al., 2023b) assess model responses using multi-agent debates. AgentVerse (Chen et al., 2023) introduces a comprehensive framework for multi-agent collaboration, highlighting emergent inter-agent behaviors. However, the alternative formats of multi-agent communication remains underexplored and NL is directly adopted across various research. Pham et al. (2023) explores agent communication with hidden states, but is limited to agents with the same open-source LLM. Our work analyze communication formats for both homogeneous and heterogeneous LLMs.

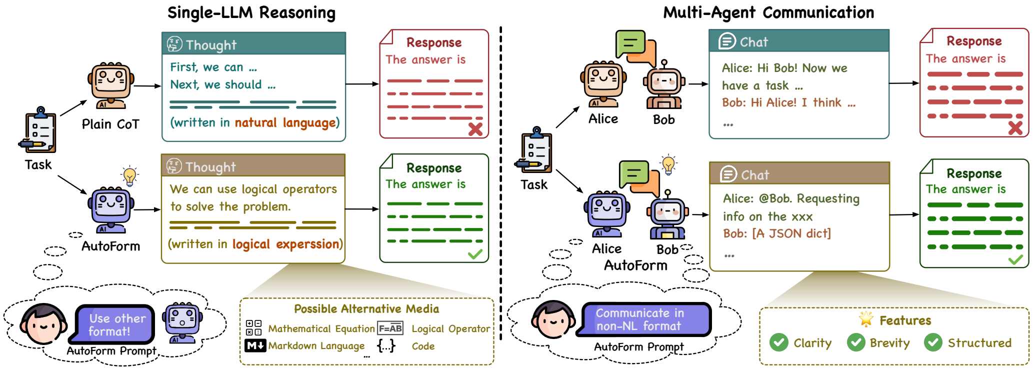


Figure 2: Overview of single-LLM reasoning and multi-agent communication using plain CoT versus the CoT with AutoForm. The left side depicts the shift from natural language to alternative formats in single-LLM reasoning, while the right side illustrates the enhanced efficiency in multi-agent communication.

3 Method

3.1 Problem Formulation

Consider an LLM parameterized by θ, denoted as p_θ. In response to a task description $\boldsymbol{x}$ and a prompt $\boldsymbol{p}$, CoT prompting initially guides the model to generate thought $\boldsymbol{t} = \{t_i\}$ utilizing a thought format m_t. While this format is often unspecified and defaults to natural language, alternative formats are feasible. The LLM then formulates an answer $\boldsymbol{y} = \{y_j\}$ based on $\boldsymbol{t}$. This process is mathematically expressed as sampling from the conditional probability distribution:

$$\begin{aligned} p_\theta(\boldsymbol{y}, \boldsymbol{t}|\boldsymbol{p}, \boldsymbol{x}, m_t) &= p_\theta(\boldsymbol{y}|\boldsymbol{t}, \boldsymbol{p}, \boldsymbol{x}, m_t) p_\theta(\boldsymbol{t}|\boldsymbol{p}, \boldsymbol{x}, m_t), \\ p_\theta(\boldsymbol{t}|\boldsymbol{p}, \boldsymbol{x}, m_t) &= \prod_i p_\theta\left(t_i|\boldsymbol{t}_{<i}, \boldsymbol{p}, \boldsymbol{x}, m_t\right), \\ p_\theta(\boldsymbol{y}|\boldsymbol{t}, \boldsymbol{p}, \boldsymbol{x}, m_t) &= \prod_j p_\theta\left(y_i|\boldsymbol{y}_{<i}, \boldsymbol{t}, \boldsymbol{p}, \boldsymbol{x}, m_t\right). \end{aligned} \tag{1}$$

In multi-agent scenarios, we extend this formulation to encompass communication among multiple LLMs, each characterized by θ_k. We consider a set of LLMs $\{p_{\theta_k}\}$ collaborating on a task. Communication among these agents utilizes format m_c, which can be NL or other alternative formats. This communication is formulated as:

$$\begin{aligned} &p_{\theta_k}(\boldsymbol{y}_k, \boldsymbol{t}|\mathcal{Y}, \boldsymbol{p}, \boldsymbol{x}, m_c, m_t) = \\ &\qquad p_{\theta_k}(\boldsymbol{y}_k|\boldsymbol{t}, \mathcal{Y}, \boldsymbol{p}, \boldsymbol{x}, m_c) \cdot p_{\theta_k}(\boldsymbol{t}|\mathcal{Y}, \boldsymbol{p}, \boldsymbol{x}, m_t), \end{aligned} \tag{2}$$

here we slightly abuse the notation and use $\boldsymbol{y}_k$ to denote the response generated by agent k, and $\mathcal{Y}$ denote the communication history.

3.2 Format Choosing for LLMs

Building upon the framework delineated in Section 3.1, our work investigates the effectiveness of allowing the LLMs to decide the thought and communication format before actually starting reasoning or communicating. At the heart of our method is the exploration of alternative formats beyond natural language. We hypothesize that various formats, such as structured data formats (e.g., JSON, markdown tables, lists) or symbolic representations (e.g., logical expressions, mathematical equations), can potentially yield more precise and effective reasoning and streamline communication.

We employ a simple yet effective prompting mechanism, where the LLMs are prompted to select and utilize the format most conducive to the task at hand, which we term as **AutoForm** (Autonomously-Decided Format). The overview of AutoForm is illustrated at Fig. 2. Specifically, for single-LLM reasoning, we add an instruction encouraging the use of non-NL formats to the original CoT prompt. In multi-agent scenarios, a similar instruction for format decision is also added. In this way, the LLMs implicitly determine and use the thought format $m_t^* = p_\theta(\boldsymbol{x}, \boldsymbol{p}_t)$ for single-LLM reasoning and the communication format $m_c^* = p_\theta(\boldsymbol{x}, \boldsymbol{p}_c)$ for multi-agent communication, where $\boldsymbol{p}_t$ and $\boldsymbol{p}_c$ include instructions for format decision.

4 Experiments

4.1 Experimental Settings

Single-LLM Reasoning. In a preliminary experiment, we prompt various LLMs to use formats other than NL for solving reasoning problems. The results, shown in Table 6, reveal significant variability in performance across different formats. For instance, GPT-3.5 exhibited a 66.8% performance gap between using an ordered list and a multi-level

Model	Logic Grid	Coin Flip	Info Essen	MM QA	AQuA	Average
GPT-3.5 CoT	46.7$_{\pm 1.6}$	23.1$_{\pm 1.0}$	32.3$_{\pm 3.2}$	24.9$_{\pm 0.8}$	60.9$_{\pm 1.2}$	41.1$_{\pm 1.8}$
+*AutoForm*	**48.0**$_{\pm 3.9}$	**39.4**$_{\pm 1.1}$	**36.7**$_{\pm 3.2}$	**26.8**$_{\pm 0.6}$	**63.7**$_{\pm 0.7}$	**46.0**$_{\pm 2.3}$
Gemini Pro CoT	49.7$_{\pm 0.2}$	47.5$_{\pm 0.2}$	34.3$_{\pm 0.7}$	28.1$_{\pm 0.7}$	56.3$_{\pm 0.6}$	43.2$_{\pm 0.5}$
+*AutoForm*	**51.2**$_{\pm 0.8}$	**57.6**$_{\pm 0.7}$	**39.2**$_{\pm 1.8}$	**31.3**$_{\pm 1.1}$	**60.0**$_{\pm 0.4}$	**47.9**$_{\pm 1.1}$
GPT-4 CoT	61.8$_{\pm 1.6}$	93.4$_{\pm 1.0}$	78.4$_{\pm 2.5}$	38.4$_{\pm 1.1}$	79.1$_{\pm 0.3}$	71.8$_{\pm 1.5}$
+*AutoForm*	**65.8**$_{\pm 2.2}$	**98.4**$_{\pm 0.2}$	**76.9**$_{\pm 2.5}$	**41.7**$_{\pm 0.9}$	**80.4**$_{\pm 0.8}$	**74.1**$_{\pm 1.6}$

Table 1: Comparative performance of single LLM reasoning across various datasets. "Information Essentiality" dataset is abbreviated as "Info Essen," and "Minute Mysteries QA" is referred to as "MM QA" for conciseness.

list, highlighting the intrinsic suitability of specific formats for different tasks.

However, in practical applications, selecting optimal formats for each task may be impractical. To address this, we conduct a comprehensively evaluate the impact of automatically chosen thought formats on LLMs' reasoning performance. We select reasoning benchmarks covering different types of reasoning, including logical reasoning (Logic Grid Puzzle (Srivastava et al., 2022) and Information Essentialy (Srivastava et al., 2022)), mathematical reasoning (AQuA (Ling et al., 2017)), causal reasoning (Minute Mysteries QA (Srivastava et al., 2022)) and symbolic reasoning (Coin Flip (Wei et al., 2022)). For all these tasks, we require the LLMs to generate the answer in a particular format, and we extract the answer with a written regular expression. The average accuracy and the standard deviation of each dataset over 3 runs are reported on most of the datasets. For more details on the experimental settings, please refer to Appendix A.

Multi-Agent Communication. To measure whether alternative formats can streamline communication, we consider scenarios where two agents with different knowledge or contexts are tasked with answering a question. The answer to the question should be derived from the knowledge of only one of the agents, or both agents' knowledge collectively, therefore requiring information exchange. The two agents speak in turn to discuss and give the final answer. To create such scenarios, we utilize three existing datasets: Hotpot QA (Yang et al., 2018), Wiki Hop (Welbl et al., 2018), and Narrative QA (Kociský et al., 2018). Hotpot QA and Wiki Hop are two multi-hop QA datasets, which require multiple sentences or paragraphs to deduce the final answer. We randomly assign the text segments provided in the datasets to two different agents. Communication is thus needed to derive the correct answer. We also explore assigning each agent part of the supporting facts, thus needing more communication to derive the answer. Narrative QA requires the model to read the whole book and answer a question. The length of a book often exceeds the context limit of the LLMs. We split the books into nearly equal sizes for the two agents and ask them to answer the question. While supporting facts in Narrative QA are not guaranteed to reside in different segments, this division still introduces critical challenges: agents must identify who possesses the necessary information, necessitating communication. For evaluation, we use F1 score as the primary metric. More details are elaborated in Appendix A.

Research Questions. To comprehensively explore the potential of LLMs in selecting suitable formats for both single-agent reasoning and multi-agent communication, we conduct an in-depth analysis addressing six pivotal research questions (RQs). These questions aim to unravel the intricacies of format selection by LLMs and its impact on task performance across various scenarios:

- **RQ1**: Can LLMs select the suitable formats autonomously? (Section 4.2)
- **RQ2**: What formats are chosen in single-LLM reasoning? (Section 4.3)
- **RQ3**: Can LLMs devise a general format for a task based on some task inputs? (Section 4.4)
- **RQ4**: Is the decided format transferable between different LLMs? (Section 4.5)
- **RQ5**: What are the features of the formats used in multi-agent communication? (Section 4.6)
- **RQ6**: Does the autonomously determined multi-agent communication format align with conventional agent communication languages such as KQML (Finin et al., 1994)? (Section 4.7)

	Wiki Hop			Hotpot QA			Narrative QA		
Model	**F1**	**# Tokens**	**ΔTokens**	**F1**	**# Tokens**	**ΔTokens**	**F1**	**# Tokens**	**ΔTokens**
GPT-4 + GPT-3.5	0.53	281.5	-	0.64	345.5	-	0.39	178.3	-
+*AutoForm*	**0.54**	**255.0**	-9.4%	**0.70**	**94.3**	-72.7%	**0.43**	**119.4**	-33.0%
GPT-4 + GPT-4	0.53	237.5	-	0.67	145.2	-	**0.43**	240.7	-
+*AutoForm*	**0.55**	**146.2**	-38.4%	**0.76**	**115.0**	-20.8%	**0.43**	**141.7**	-41.1%

Table 2: Comparative performance in multi-agent communication across various QA datasets. The table highlights RougeL scores, with better performance in different model pairing settings indicated in bold. The **Δ**Tokens column quantifies the token reduction achieved by the AutoForm method.

4.2 RQ1: The Capability of LLMs on Selecting Suitable Format

Single-LLM Reasoning. The comparative efficacy of the AutoForm approach over the conventional Chain-of-Thought (CoT) methodology in single-LLM reasoning tasks is encapsulated in Table 1. We observe clear performance improvements when employing AutoForm across different datasets, compared to the baseline CoT method.

For GPT-3.5, the implementation of AutoForm leads to a significant improvement in accuracy, particularly notable in the Coin Flip dataset, where accuracy escalates from 22.2% to 38.0%. This substantial increase highlights the model's enhanced ability in symbolic reasoning. Across other datasets, AutoForm yields consistent enhancements, with increases generally ranging between 3% to 5%, culminating in an overall average performance boost of 5.4%. Similarly, for Gemini Pro, AutoForm achieves an average performance enhancement of 5.7%. GPT-4 also benefits from AutoForm, with an average performance uplift of 3.3% across all tasks. These consistent improvements across various datasets demonstrate the method's model-agnostic robustness and the efficacy of utilizing alternative formats in reasoning tasks. It also suggests that alternative formats, apart from NL, can aid LLMs in task resolution. It is just that without explicit reminders, the LLMs do not explore alternative formats.

Multi-Agent Communication. The outcomes of our multi-agent communication experiments, detailed in Table 2, provide valuable insights into the efficiency and effectiveness of utilizing alternative communication formats in collaborative environments. In this experiment, we experiment with different model pairings to explore this robustness more comprehensively. Since the initial speaking agent often sets the tone for the communication format, we vary the speaking order in pairings of heterogeneous models, e.g., *GPT-4 + GPT-3.5* in the table indicates GPT-4 is the initiator. Due to page limit, we place results where GPT-3.5 initiates the conversation, and where the supporting facts are split and distributed to two agents at Appendix B.

A key finding from our experiments is the notable efficiency achieved through AutoForm, as evidenced by the substantial reduction in token usage across diverse model pairings and speaking orders. This efficiency, quantified in the **Δ**Tokens column, illustrates the capability of LLMs to move beyond their typical NL to adopt more concise and efficient communication formats. This is particularly evident in the Hotpot QA dataset with the GPT-4 and GPT-3.5 pairing, where we witness a token reduction reaching 72.7%. On most of the other datasets, AutoForm also obtains substantial token reduction. These findings imply that LLMs, though extensively trained on NL, are capable of exploring and employing alternative formats to enhance communication efficiency. Detailed case studies further elucidating the features of the chosen formats will be presented in Section 4.6.

Furthermore, the effectiveness of the multi-agent communication facilitated by AutoForm, as gauged by F1 scores, is found to be largely comparable to, and occasionally exceeding, that of natural language-based interactions. This is especially true when GPT-4 initiates the conversation, suggesting that more advanced LLMs possess a better ability to select communication formats that strike a balance between conciseness and clarity. Conversely, we show in Appendix B that with GPT-3.5 as the initiator, despite the similar notable efficiency in token usage, the performance is generally akin to or slightly below that achieved with natural language. These observations highlight the intricate challenge of selecting an appropriate communication format, a task that proves demanding even for sophisticated LLMs, and the importance of balancing brevity with the need to preserve the integrity of the communicative content.

4.3 RQ2: Formats Chosen in Single-LLM Reasoning

In addressing RQ1, we investigate the formats selected from LLMs when tasked with reasoning. This analysis is pivotal in understanding how LLMs, when granted the autonomy to choose, navigate away from the default NL format to potentially more efficient alternatives.

We analyze a randomly sampled set of 50 examples from each dataset, investigate the reasoning traces produced with AutoForm, and manually count the number of appeared formats. Fig. 3(a-c) display the distribution of formats chosen by Gemini Pro, GPT-3.5 and GPT-4, and Fig. 3(d) displays their combined preferences across various tasks. The data indicates a notable diversity in format selection by LLMs. A shift towards structured formats, such as lists, logical expressions, and markdown tables is observed. These formats are particularly favored in tasks that demand logical reasoning, offering clearer and more concise data representation, as illustrated in Fig. 3(d).

4.4 RQ3: Generalization of Format Selection Based on Task Inputs

An examination of Fig. 3(d) reveals a discernible variation in the LLMs' format preferences across different tasks. This variation aligns with the expectation that the optimal format would naturally differ between tasks, each with its unique requirements. In addressing RQ2, we probe whether LLMs are capable of identifying a general format suitable for a given task based on a subset of inputs, and then consistently applying this format for problem-solving. In AutoForm, as delineated in Section 3.2, LLMs typically select a format *implicitly* for each instruction on a case-by-case basis. Nonetheless, it stands to reason that certain tasks may be inherently conducive to a specific format. To investigate this hypothesis, we introduce the **two-step AutoForm**. This approach tasks an LLM with first determining the most efficient format and subsequently utilizing that format in the CoT problem-solving stage. That is, we turn the implicit format decision into an explicit step, which mathematically follows Eq. (1) instead of merging the two steps.

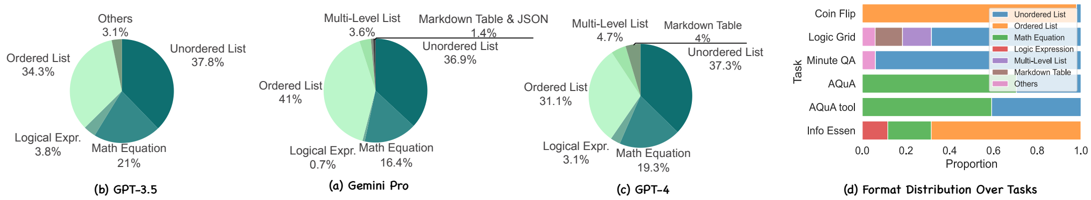
The two-step AutoForm experiments with two distinct settings: 1) *Instance-Based*, where the LLM selects a format for each instruction, and 2) *Task-Based*, where the LLM deduces a general format for the entire task by analyzing 5 inputs from the task. Note that, unlike few-shot prompting, the Task-Based setting does not provide answers within the inputs, and these inputs are only utilized during the format decision step.

The results are detailed in Table 3. The Task-Based setting demonstrates that both GPT-3.5 and Gemini Pro can effectively generalize a thought format from a limited set of inputs within a task, and often outperforming the Instance-Based setting. In contrast, GPT-4 shows similar performance levels in both Task-Based and Instance-Based settings, suggesting that its advanced capabilities may afford it greater flexibility in format usage. These findings indicate that LLMs, particularly less sophisticated ones like GPT-3.5, may benefit from exposure to multiple inputs from a task to better generalize an effective thought format. This ability of LLMs to generalize the format for a task makes the AutoForm approach more practical since the format can be identified only once for a specific task.

4.5 RQ4: Transferability of Format Across Different LLMs

This subsection delves into RQ3, the transferability of the format across LLMs. The concept of format transferability is crucial in understanding the universality of formats decisions made by LLMs and their applicability across various models. Instead of using the same model for format selection and problem-solving (homogeneous setting) as we have done in Section 4.4, we explore the heterogeneous setting of the two-step AutoForm, where different models are employed for the two steps.

The results are presented in the last group of Table 3. Generally, the format is transferable, but may lead to slightly inferior performances compared to the homogeneous setting. For example, transferring the format decided by GPT-4 to GPT-3.5 or Gemini Pro leads to a decrease compared to the homogeneous setting for the two LLMs in most of the tasks except for Coin Flip. On the other hand, when transferring the format decided by GPT-3.5 to GPT-4, the results are generally comparable to the homogeneous setting for GPT-4. For information essentiality, the format selected by GPT-3.5 is generally less efficient, leading to incomplete problem-solving processes. For other tasks, the format generated by GPT-3.5 proves adequate and is easily interpreted by GPT-4, resulting in simi-



(b) GPT-3.5

(a) Gemini Pro

(c) GPT-4

(d) Format Distribution Over Tasks

Figure 3: Format distribution chosen by Gemini Pro (a), GPT-3.5 (b) and GPT-4 (c), and the overall format distribution across tasks from both models (d).

Model	Logic Grid	Coin Flip	Info Essen	MM QA	AQuA	Average
GPT-3.5 (Instance-Based)	48.0	49.2	39.7	26.1	**66.5**	45.9$_{(+7.5\%)}$
GPT-3.5 (Task-Based)	**51.0**	**62.8**	**42.6**	**28.1**	65.0	**49.9**$_{(+11.5\%)}$
Gemini Pro (Instance-Based)	39.5	44.8	**35.3**	27.1	**59.4**	41.2$_{(+2.5\%)}$
Gemini Pro (Task-Based)	**41.5**	**47.8**	**35.3**	**28.6**	**59.4**	**42.5**$_{(+3.8\%)}$
GPT-4 (Instance-Based)	**71.5**	**100.0**	**76.5**	**41.4**	78.3	**73.5**$_{(+3.2\%)}$
GPT-4 (Task-Based)	70.0	99.8	75.0	**41.4**	**79.5**	73.1$_{(+2.8\%)}$
GPT-4 ⇒ GPT-3.5 (Task-Based)	47.5	83.0	35.3	23.2	59.4	49.7$_{(+11.2\%)}$
GPT-4 ⇒ Gemini Pro (Task-Based)	40.0	74.6	25.0	25.6	47.2	42.5$_{(+3.8\%)}$
GPT-3.5 ⇒ GPT-4 (Task-Based)	65.5	98.6	72.1	42.9	79.5	71.7$_{(+1.3\%)}$

Table 3: Comparative performance of two-step AutoForm with single LLM reasoning across various datasets. The notation $model_1 \Rightarrow model_2$ denotes using $model_1$ for format selection, and $model_2$ for problem-solving. Average performance improvements over CoT results, as presented in Table 1, are denoted with a subscript.

lar performances to the homogeneous setting for GPT-4.

4.6 RQ5: Features of Communication Format

In addressing RQ4, this subsection investigates the characteristics of the formats used by language agents in multi-agent communication scenarios. Our goal is to identify key attributes contributing to efficient communication by examining the formats used during their interactions.

We analyze 50 random interaction logs for each dataset and present some cases in Fig. 4. Despite some retained characteristics of NL, the communication formats decided via AutoForm display distinct features:

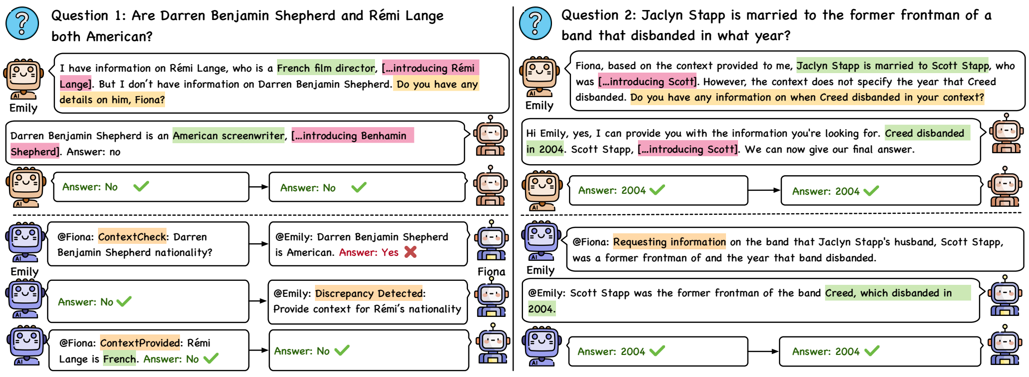
Clarity and Structure. An important feature of the selected formats is an emphasis on clarity. LLMs consistently favor formats facilitating unambiguous and straightforward communication, which is vital in our multi-agent scenarios. In these scenarios, agents possess divergent knowledge sets, making the clear exchange of this distinct information indispensable. Structured formats, which provide an organized method of presenting information, are also prevalent. These formats enhance the comprehensibility and accessibility of the content. Contrasting this with the use of NL, we observe that the LLM-decided formats tend to be more direct and clear, effectively reducing redundancy.

Brevity and Efficiency. Another key feature is the focus on brevity, enhancing communication efficiency. Formats chosen by LLMs often omit elements like pleasantries or emotive expressions, resulting in concise exchanges. This brevity conserves computational resources and concentrates dialogue on the task at hand, optimizing the communication process for faster, more efficient information exchange. This is particularly beneficial in scenarios requiring rapid and effective decision-making.

4.7 RQ6: Alignment with Conventional Agent Communication Languages

RQ5 probes the extent to which the communication formats determined by LLMs using AutoForm align with traditional Agent Communication Languages (ACLs), such as KQML (Finin et al., 1994) and FIPA-ACL (FIPA, 2001). These ACLs have been instrumental in structuring communication between intelligent agents to foster cooperation and coordination. A typical KQML message, as shown below, exemplifies the structured nature of traditional ACLs:

```
(ask-one
  :sender joe
  :content (PRICE IBM ?price)
  :receiver stock-server
```



Question 1: Are Darren Benjamin Shepherd and Rémi Lange both American?

I have information on Rémi Lange, who is a French film director, [...introducing Rémi Lange]. But I don't have information on Darren Benjamin Shepherd. Do you have any details on him, Fiona?

Darren Benjamin Shepherd is an American screenwriter, [...introducing Benhamin Shepherd]. Answer: no

Answer: No ✓

Answer: No ✓

@Fiona: ContextCheck: Darren Benjamin Shepherd nationality?

@Emily: Darren Benjamin Shepherd is American. Answer: Yes ✗

Answer: No ✓

@Emily: Discrepancy Detected: Provide context for Rémi's nationality

@Fiona: ContextProvided: Rémi Lange is French. Answer: No ✓

Answer: No ✓

(a) Clarity and Structured

Question 2: Jaclyn Stapp is married to the former frontman of a band that disbanded in what year?

Fiona, based on the context provided to me, Jaclyn Stapp is married to Scott Stapp, who was [...introducing Scott]. However, the context does not specify the year that Creed disbanded. Do you have any information on when Creed disbanded in your context?

Hi Emily, yes, I can provide you with the information you're looking for. Creed disbanded in 2004. Scott Stapp, [...introducing Scott]. We can now give our final answer.

Answer: 2004 ✓

Answer: 2004 ✓

@Fiona: Requesting information on the band that Jaclyn Stapp's husband, Scott Stapp, was a former frontman of and the year that band disbanded.

@Emily: Scott Stapp was the former frontman of the band Creed, which disbanded in 2004.

Answer: 2004 ✓

Answer: 2004 ✓

(b) Brevity and Efficiency

Figure 4: Multi-agent communication examples. The top panel illustrates a traditional natural language conversation, and the bottom panel shows a conversation using AutoForm. Necessary information related to the question is marked in green , redundant information is marked in red , and speech-act-related phrases are marked in orange .

```
  :reply-with ibm-stock
  :language LPROLOG
  :ontology NYSE-TICKS)
```

Our examination of the communication patterns emerging from AutoForm reveals an interesting resemblance to these structured elements. As depicted in Fig. 4, LLMs frequently employ a structured format where "@" denotes the *receiver*, verb phrases such as "ContextCheck" indicate the *performative* (the *"ask-one"* in the above example), and a succinct text string encapsulates the *content*. This structuring mirrors the composition of ACL messages, where each part serves a specific function in the communication process.

Intrigued by this similarity, we conduct an experiment where LLMs are prompted to communicate using a format similar to KQML:

```
(performative
  :content ...
  :receiver ...)
```

The results of this experiment on two GPT-4-based agents are presented in Table 4. We have two settings, one prompts the LLMs to use the exact KQML format as presented above, and the other setting uses the JSON version of KQML format, considering that LLMs may be more adept at JSON. While both settings show worse or comparable performance to AutoForm in terms of F1, the number of tokens consumed is larger than AutoForm. This finding suggests that while LLMs can indeed emulate the formality of traditional ACL formats, the AutoForm approach optimizes the communication by enhancing clarity and structure, yet concurrently reduces token usage.

	Hotpot QA		Wiki Hop		Narrative QA	
Format	**F1**	**#Tokens**	**F1**	**#Tokens**	**F1**	**#Tokens**
KQML	**0.74**	313.8	0.53	368.1	0.27	343.3
JSON	0.69	346.0	**0.54**	291.4	0.21	385.2
AutoForm	**0.74**	**115.0**	0.53	**146.2**	**0.43**	**141.7**

Table 4: Multi-agent communication performances using conventional ACL format.

These results highlight two key implications. First, AutoForm can generate communication patterns similar to established ACLs. Second, it efficiently distills these traditional formats into a more concise form, conserving computational resources while maintaining communicative effectiveness. This balance of clarity, structure, and brevity makes AutoForm a powerful tool for facilitating intelligent agent communication in various contexts.

5 Conclusion

In this work, we demonstrate that LLMs can autonomously determine suitable non-NL formats for reasoning and communication using the AutoForm prompting method. Our analyses address six key research questions, showing that LLMs can generalize a reasoning format from task-specific examples and transfer it across different models. Additionally, the communication formats generated by LLMs resemble traditional ACLs, offering both precision and efficiency. These insights enhance our understanding of LLMs' capabilities beyond NL, improving LLM reasoning and inter-agent communication.

Limitations

Despite we have shown many kinds of formats can facilitate LLMs reasoning and communication, the scope of alternative formats explored is still not exhaustive. The potential of numerous other formats and their specific applications to various LLM architectures warrants further investigation.

Moreover, the generalization of chosen formats across tasks, while promising, shows variability in effectiveness depending on the complexity of the task and the specific LLM used. This variability highlights the nuanced nature of format suitability and its impact on task performance, suggesting that further exploration is necessary to fully harness the capabilities of alternative formats.

Acknowledgement

The work was supported by the National Key R&D Program of China No.2022ZD0116312), the Postdoctoral Fellowship Program of CPSF under Grant Number GZB20230348, Tencent Rhino-Bird Focused Research Program, and the Young Elite Scientists Sponsorship Program by CAST (2023QNRC001).

References

Anthropic. 2023. Introducing claude 2.1.

Daniil A. Boiko, Robert MacKnight, and Gabe Gomes. 2023. Emergent autonomous scientific research capabilities of large language models. *CoRR*, abs/2304.05332.

Chi-Min Chan, Weize Chen, Yusheng Su, Jianxuan Yu, Wei Xue, Shanghang Zhang, Jie Fu, and Zhiyuan Liu. 2023. Chateval: Towards better llm-based evaluators through multi-agent debate. *CoRR*, abs/2308.07201.

Weize Chen, Yusheng Su, Jingwei Zuo, Cheng Yang, Chenfei Yuan, Chen Qian, Chi-Min Chan, Yujia Qin, Yaxi Lu, Ruobing Xie, Zhiyuan Liu, Maosong Sun, and Jie Zhou. 2023. Agentverse: Facilitating multi-agent collaboration and exploring emergent behaviors in agents. *CoRR*, abs/2308.10848.

Wenhu Chen, Xueguang Ma, Xinyi Wang, and William W. Cohen. 2022. Program of thoughts prompting: Disentangling computation from reasoning for numerical reasoning tasks. *CoRR*, abs/2211.12588.

Noam Chomsky. 2006. *Language and mind*. Cambridge University Press.

Yilun Du, Shuang Li, Antonio Torralba, Joshua B. Tenenbaum, and Igor Mordatch. 2023. Improving factuality and reasoning in language models through multiagent debate. *CoRR*, abs/2305.14325.

Timothy W. Finin, Richard Fritzson, Donald P. McKay, and Robin McEntire. 1994. KQML as an agent communication language. In *Proceedings of the Third International Conference on Information and Knowledge Management (CIKM'94), Gaithersburg, Maryland, USA, November 29 - December 2, 1994*, pages 456–463. ACM.

FIPA. 2001. *FIPA ACL Message Structure Specification*. FIPA.

Jerry A Fodor. 1975. *The language of thought*, volume 5. Harvard university press.

Luyu Gao, Aman Madaan, Shuyan Zhou, Uri Alon, Pengfei Liu, Yiming Yang, Jamie Callan, and Graham Neubig. 2023. PAL: program-aided language models. In *International Conference on Machine Learning, ICML 2023, 23-29 July 2023, Honolulu, Hawaii, USA*, volume 202 of *Proceedings of Machine Learning Research*, pages 10764–10799. PMLR.

Gemini Team Google, Rohan Anil, Sebastian Borgeaud, Yonghui Wu, Jean-Baptiste Alayrac, Jiahui Yu, Radu Soricut, Johan Schalkwyk, Andrew M Dai, Anja Hauth, et al. 2023. Gemini: a family of highly capable multimodal models. *arXiv preprint arXiv:2312.11805*.

Sirui Hong, Xiawu Zheng, Jonathan Chen, Yuheng Cheng, Jinlin Wang, Ceyao Zhang, Zili Wang, Steven Ka Shing Yau, Zijuan Lin, Liyang Zhou, Chenyu Ran, Lingfeng Xiao, and Chenglin Wu. 2023. Metagpt: Meta programming for multi-agent collaborative framework. *CoRR*, abs/2308.00352.

Tomás Kociský, Jonathan Schwarz, Phil Blunsom, Chris Dyer, Karl Moritz Hermann, Gábor Melis, and Edward Grefenstette. 2018. The narrativeqa reading comprehension challenge. *Trans. Assoc. Comput. Linguistics*, 6:317–328.

Takeshi Kojima, Shixiang Shane Gu, Machel Reid, Yutaka Matsuo, and Yusuke Iwasawa. 2022. Large language models are zero-shot reasoners. In *NeurIPS*.

George Lakoff. 2008. *Women, fire, and dangerous things: What categories reveal about the mind*. University of Chicago press.

Guohao Li, Hasan Abed Al Kader Hammoud, Hani Itani, Dmitrii Khizbullin, and Bernard Ghanem. 2023a. CAMEL: communicative agents for "mind" exploration of large scale language model society. *CoRR*, abs/2303.17760.

Ruosen Li, Teerth Patel, and Xinya Du. 2023b. PRD: peer rank and discussion improve large language model based evaluations. *CoRR*, abs/2307.02762.

Wang Ling, Dani Yogatama, Chris Dyer, and Phil Blunsom. 2017. Program induction by rationale generation: Learning to solve and explain algebraic word problems. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics, ACL 2017, Vancouver, Canada, July 30 - August 4, Volume 1: Long Papers*, pages 158–167. Association for Computational Linguistics.

Tengxiao Liu, Qipeng Guo, Yuqing Yang, Xiangkun Hu, Yue Zhang, Xipeng Qiu, and Zheng Zhang. 2023. Plan, verify and switch: Integrated reasoning with diverse x-of-thoughts. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing, EMNLP 2023, Singapore, December 6-10, 2023*, pages 2807–2822. Association for Computational Linguistics.

Grégoire Mialon, Clémentine Fourrier, Craig Swift, Thomas Wolf, Yann LeCun, and Thomas Scialom. 2023. GAIA: a benchmark for general AI assistants. *CoRR*, abs/2311.12983.

OpenAI. 2023a. Assistants api.

OpenAI. 2023b. GPT-4 technical report. *CoRR*, abs/2303.08774.

Anton Osika. 2023. gpt-engineer.

Joon Sung Park, Joseph C. 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 2023, San Francisco, CA, USA, 29 October 2023- 1 November 2023*, pages 2:1–2:22. ACM.

Chau Pham, Boyi Liu, Yingxiang Yang, Zhengyu Chen, Tianyi Liu, Jianbo Yuan, Bryan A. Plummer, Zhaoran Wang, and Hongxia Yang. 2023. Let models speak ciphers: Multiagent debate through embeddings. *CoRR*, abs/2310.06272.

Steven Pinker. 2003. *The language instinct: How the mind creates language*. Penguin uK.

Chen Qian, Xin Cong, Cheng Yang, Weize Chen, Yusheng Su, Juyuan Xu, Zhiyuan Liu, and Maosong Sun. 2023. Communicative agents for software development. *CoRR*, abs/2307.07924.

Noah Shinn, Beck Labash, and Ashwin Gopinath. 2023. Reflexion: an autonomous agent with dynamic memory and self-reflection. *CoRR*, abs/2303.11366.

Significant Gravitas. AutoGPT.

Aarohi Srivastava, Abhinav Rastogi, Abhishek Rao, Abu Awal Md Shoeb, Abubakar Abid, Adam Fisch, Adam R. Brown, Adam Santoro, Aditya Gupta, Adrià Garriga-Alonso, Agnieszka Kluska, Aitor Lewkowycz, Akshat Agarwal, Alethea Power, Alex Ray, Alex Warstadt, Alexander W. Kocurek, Ali Safaya, Ali Tazarv, Alice Xiang, Alicia Parrish, Allen Nie, Aman Hussain, Amanda Askell, Amanda Dsouza, Ameet Rahane, Anantharaman S. Iyer, Anders Andreassen, Andrea Santilli, Andreas Stuhlmüller, Andrew M. Dai, Andrew La, Andrew K. Lampinen, Andy Zou, Angela Jiang, Angelica Chen, Anh Vuong, Animesh Gupta, Anna Gottardi, Antonio Norelli, Anu Venkatesh, Arash Gholamidavoodi, Arfa Tabassum, Arul Menezes, Arun Kirubarajan, Asher Mullokandov, Ashish Sabharwal, Austin Herrick, Avia Efrat, Aykut Erdem, Ayla Karakas, and et al. 2022. Beyond the imitation game: Quantifying and extrapolating the capabilities of language models. *CoRR*, abs/2206.04615.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Brian Ichter, Fei Xia, Ed H. Chi, Quoc V. Le, and Denny Zhou. 2022. Chain-of-thought prompting elicits reasoning in large language models. In *NeurIPS*.

Johannes Welbl, Pontus Stenetorp, and Sebastian Riedel. 2018. Constructing datasets for multi-hop reading comprehension across documents. *Trans. Assoc. Comput. Linguistics*, 6:287–302.

Benjamin Lee Whorf. 2012. *Language, thought, and reality: Selected writings of Benjamin Lee Whorf*. MIT press.

Qingyun Wu, Gagan Bansal, Jieyu Zhang, Yiran Wu, Shaokun Zhang, Erkang Zhu, Beibin Li, Li Jiang, Xiaoyun Zhang, and Chi Wang. 2023. Autogen: Enabling next-gen LLM applications via multi-agent conversation framework. *CoRR*, abs/2308.08155.

Zhilin Yang, Peng Qi, Saizheng Zhang, Yoshua Bengio, William W. Cohen, Ruslan Salakhutdinov, and Christopher D. Manning. 2018. Hotpotqa: A dataset for diverse, explainable multi-hop question answering. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing, Brussels, Belgium, October 31 - November 4, 2018*, pages 2369–2380. Association for Computational Linguistics.

Shunyu Yao, Dian Yu, Jeffrey Zhao, Izhak Shafran, Thomas L. Griffiths, Yuan Cao, and Karthik Narasimhan. 2023a. Tree of thoughts: Deliberate problem solving with large language models. *CoRR*, abs/2305.10601.

Shunyu Yao, Jeffrey Zhao, Dian Yu, Nan Du, Izhak Shafran, Karthik R. Narasimhan, and Yuan Cao. 2023b. React: Synergizing reasoning and acting in language models. In *The Eleventh International Conference on Learning Representations, ICLR 2023, Kigali, Rwanda, May 1-5, 2023*. OpenReview.net.

Shuyan Zhou, Frank F. Xu, Hao Zhu, Xuhui Zhou, Robert Lo, Abishek Sridhar, Xianyi Cheng, Yonatan Bisk, Daniel Fried, Uri Alon, and Graham Neubig. 2023. Webarena: A realistic web environment for building autonomous agents. *CoRR*, abs/2307.13854.

A Experimental Settings

In this section, we introduce the details of our experimental settings.

A.1 Models

For OpenAI's models, we use gpt-3.5-turbo-1106 and gpt-4-1106-preview. For Gemini pro, we use the Gemini pro 1.0 in 2024.1.

A.2 Dataset Pre-Processing

The statistics of the processed data are presented at Table 5. We now elaborate the dataset download and pre-process process.

Single-LLM Reasoning. For Logic Grid, Information Essentiality and Minute Mysteries QA that are from Big-Bench, we download the dataset from the official repo[1]. For Coin Flip, we download the dataset from `https://huggingface.co/datasets/skrishna/coin_flip`, and use the first 500 examples in the test set. For AQuA, we download the dataset from `https://huggingface.co/datasets/aqua_rat/` and use its test set.

Multi-Agent Communication. For the Hotpot QA dataset, we adhere to the methodology outlined by Reflexion (Shinn et al., 2023), obtaining the dataset from their repository [2]. In the case of Wiki Hop, we acquire it through the Huggingface Datasets platform, from which we randomly selected 100 examples from its validation set for our study. For the Narrative QA dataset, also sourced from Huggingface Datasets, we note inconsistencies in the quality of the e-books included. To ensure higher data quality, we exclusively utilize e-books from Project Gutenberg by checking whether the e-book starts with "Project Gutenberg's". Furthermore, considering the context length limitation of 16k tokens in GPT-3.5, we exclude e-books exceeding 30k tokens. This exclusion is to enable splitting the content into two segments, each fitting within the GPT-3.5 context limit. From this refined dataset, we randomly chose 100 examples for analysis.

A.3 Metrics

For single-LLM reasoning, we report the accuracy by comparing the generated answer with the label. For multi-agent communication, we report the F1 score of the generated answer.

B Additional Experimental Results for Multi-Agent Communication

Due to constraints on the paper length, the comprehensive experimental results, including those where GPT-3.5 serves as the initiating agent, are detailed in Table 8. Our analysis reveals that the performance of AutoForm tends to be suboptimal when compared to the baseline in scenarios initiated by GPT-3.5. A closer examination of these interactions indicates that GPT-3.5, when merely prompted to employ non-NL formats without additional guidance, frequently produces overly succinct responses, or simply gives a hallucinated answer, resulting in diminished performance. This observation underscores the need for further research into optimizing prompts for less advanced LLMs to effectively utilize non-NL formats for communication, representing a promising avenue for future exploration.

Additionally, we conduct experiments with a specific setting for HotpotQA to evaluate the performance of LLMs when supporting facts are distributed between two agents. This setting, termed *separate context*, ensures that the supporting facts are divided and distributed between the agents, necessitating effective communication to aggregate the information and derive the correct answer.

The results presented in Table 7 reveal significant insights into the performance of LLMs under the *separate context* setting. The application of the AutoForm mechanism shows a distinct impact on both models. For GPT-3.5, the RougeL score slightly decreases from 0.62 to 0.53, with a corresponding 22.4% reduction in the number of tokens generated. In contrast, the GPT-4 models exhibit an increase in performance with the AutoForm mechanism, achieving a RougeL score of 0.69 and a 33.8% reduction in token usage. This improvement highlights the effectiveness of AutoForm in not only maintaining but enhancing the performance of more advanced models while optimizing resource utilization.

[1] `https://github.com/google/BIG-bench`

[2] `https://github.com/noahshinn/reflexion/tree/main/hotpotqa_runs/data`

Dataset	# Examples	Input	Output	Category	License
Single-LLM Reasoning					
Logic Grid Puzzle	200	Clues + Question	Number	Logical Reasoning	Apache License 2.0
Information Essentiality	68	Question + Statement Options	Option Number	Logical Reasoni	Apache License 2.0
AQuA	254	Mathematical Question + Options	Option Number	Mathematical Reasoning	Apache License 2.0
Minute Mysteries QA	203	Story + Question + Options	Option Number	Causal Reasoning	Apache License 2.0
Coin Flip	500	Action Sequence	Yes / No	Symbolic Reasoning	MIT License
Multi-LLM Communication					
HotPot QA	100	Passages + Question	Free Text	Multi-Hop QA	CC BY-SA 4.0
Wiki Hop	100	Sentences + Question	Free Text	Multi-Hop QA	CC BY-SA 3.0
Narrative QA	100	Book + Question	Free Text	QA	Apache License 2.0

Table 5: The datasets we use in our experiments.

Dataset	Model	Math Equation	Unordered List	Ordered List	Markdown Table	Multi-level List	Logical Expression
Logic Grid	GPT-3.5	**50.0**	47.0	46.0	48.5	45.0	41.0
	GPT-4	58.0	**65.0**	59.0	55.0	62.0	59.0
	Gemini Pro	42.0	**50.5**	48.0	49.0	48.0	49.0
Coin Flip	GPT-3.5	71.0	39.0	**86.0**	66.0	19.2	48.0
	GPT-4	75.0	95.0	**100.0**	99.0	98.0	71.4
	Gemini Pro	47.0	61.6	60.4	**63.4**	56.4	59.6
Info Essen	GPT-3.5	22.0	**30.8**	**30.8**	29.4	**30.8**	**30.8**
	GPT-4	73.5	73.5	**76.4**	73.5	**76.4**	75.0
	Gemini Pro	**45.6**	29.4	32.4	29.4	44.1	38.2
MM QA	GPT-3.5	22.2	22.2	23.2	**27.1**	22.7	20.2
	GPT-4	39.9	41.9	37.7	36.9	**42.4**	37.9
	Gemini Pro	**29.6**	27.6	25.6	27.6	28.1	24.1
AQuA	GPT-3.5	63.4	62.2	63.0	56.7	59.8	**66.0**
	GPT-4	76.4	72.4	**80.3**	78.0	78.0	79.8
	Gemini Pro	56.7	57.5	**59.8**	47.2	55.5	56.3

Table 6: Effectiveness of various formats across different models and datasets.

	Hotpot QA$_{\text{separate context}}$		
Model	**F1**	**# Tokens**	**ΔTokens**
GPT-3.5 + GPT-3.5	**0.62**	369.3	-
+*AutoForm*	0.53	286.6	-22.4%
GPT-4 + GPT-4	0.65	151.0	-
+*AutoForm*	**0.69**	100.0	-33.8%

Table 7: Performance on HotpotQA under the *separate context* setting. The supporting facts are distributed between two agents, necessitating inter-agent communication for aggregating information and deriving the correct answer.

	Wiki Hop			Hotpot QA			Narrative QA		
Model	**RougeL**	**# Tokens**	**ΔTokens**	**RougeL**	**# Tokens**	**ΔTokens**	**RougeL**	**# Tokens**	**ΔTokens**
GPT-3.5 + GPT-3.5	**0.53**	192.6	-	**0.53**	499.7	-	**0.31**	140.0	-
+*AutoForm*	0.49	163.9	-14.9%	0.48	236.1	-52.8%	0.30	35.5	-74.6%
GPT-3.5 + GPT-4	0.56	246.8	-	**0.71**	333.9	-	**0.33**	208.8	-
+*AutoForm*	**0.57**	200.3	-18.8%	0.62	102.3	-69.4%	0.26	125.4	-39.9%
GPT-4 + GPT-3.5	0.53	281.5	-	0.64	345.5	-	0.39	178.3	-
+*AutoForm*	**0.54**	**255.0**	-9.4%	**0.70**	**94.3**	-72.7%	**0.43**	**119.4**	-33.0%
GPT-4 + GPT-4	0.53	237.5	-	0.67	145.2	-	**0.43**	240.7	-
+*AutoForm*	**0.55**	**146.2**	-38.4%	**0.76**	**115.0**	-20.8%	**0.43**	**141.7**	-41.1%

Table 8: Comparative performance in multi-agent communication across various QA datasets. The table highlights RougeL scores, with better performance in different model pairing settings indicated in bold. The ΔTokens column quantifies the token reduction achieved by the AutoForm method.

PROMPT FOR COIN FLIP

CoT:

Question:
${task_description}

At the end of your response, you must give your answer in the form of "the answer is: no" or "the answer is: yes". Let's think step-by-step.

AutoForm:

Question:
${task_description}

To enhance clarity and eliminate ambiguities inherent in natural language, consider employing more structured and concise forms of communication for your step-by-step solutions. Suitable formats include code, pseudocode, JSON, markdown tables, logical operators, or mathematical equations.

At the end of your response, you must give your answer in the form of "the answer is: no" or "the answer is: yes". Remember to be concise and accurate.

Table 9: Prompt for Coin Flip

PROMPT FOR LOGIC GRID

CoT

—
${task_description}
—

At the end of your response, you must give your answer in the form of "the answer is: {number}", where {number} is the answer number. Now solve the problem step-by-step. Use as few words as possible.

AutoForm

—
${task_description}
—

To enhance clarity and eliminate ambiguities inherent in natural language, consider employing more structured and concise forms of communication for your step-by-step solutions. Suitable formats include code, pseudocode, JSON, markdown tables, logical operators, or mathematical equations.

At the end of your response, you must give your answer in the form of "the answer is: {number}", where {number} is the answer number. Remember to be concise and accurate.

Table 10: Prompt for Logic Grid

PROMPT FOR MINUTE MYSTERIES QA

CoT:

—
${task_description}
—

Now solve the problem step-by-step. At the end of your response, you must give your answer in the form of "the correct option is: number", where number is the index of the chosen option.

AutoForm:

—
${task_description}
—

To enhance clarity and eliminate ambiguities inherent in natural language, consider employing more structured and concise forms of communication for your step-by-step solutions. Suitable formats include code, pseudocode, JSON, markdown tables, logical operators, or mathematical equations.

Now solve the problem step-by-step. At the end of your response, you must give your answer in the form of "the correct option is: number", where number is the index of the chosen option.

Table 11: Prompt for Minute Mysteries QA

PROMPT FOR AQUA

CoT:

Solve the problem presented below:
—
${task_description}
—

RESPONSE GUIDELINES:
1. Think step by step.
2. Concluding with the Answer: End your response with "Answer: {answer}", where {answer} is the final result of your problem-solving process. The {answer} should be a single capital letter.

AutoForm:

Solve the problem presented below:
—
${task_description}
—

RESPONSE GUIDELINES:
1. Initial State Representation: Begin by providing a clear and detailed representation of the initial state or conditions of the problem.
2. Step-by-Step Solution Process: Progressively update the state representation as you work through each step of the solution. This should include all logical reasoning and calculations leading to the final answer.
3. Concluding with the Answer: End your response with "Answer: {answer}", where {answer} is the final result of your problem-solving process. The {answer} should be a single capital letter.

Table 12: Prompt for AQuA

PROMPT FOR INFORMATION ESSENTIALITY

3.5+CoT:
Solve the problem presented below:
—
${task_description}
—

RESPONSE GUIDELINES:
1. Think step by step.
2. Your answer should be ended with "Answer: {answer}" where {answer} is the answer to the problem.

3.5+AutoForm:
Solve the problem presented below:
—
${task_description}
—

RESPONSE GUIDELINES:
1. Initial State Representation: Begin by providing a clear and detailed representation of the initial state or conditions of the problem.
2. Step-by-Step Solution Process:Progressively update the state representation as you work through each step of the solution. This should include all logical reasoning and calculations leading to the final answer.
3. To enhance clarity and eliminate ambiguities inherent in natural language, consider employing more structured and concise forms of communication for your step-by-step solutions. Suitable formats include code, pseudocode, JSON, markdown tables, logical operators, or mathematical equations.
4. Concluding with the Answer: End your response with "Answer: {answer}", where {answer} is the final result of your problem-solving process.

4+CoT:
Solve the problem presented below:
—
${task_description}
—

RESPONSE GUIDELINES:
1.You should think step by step.
2. You should consider three scenarios: using only Statement 1, using only Statement 2, and using both Statements.
3. Note (IMPORTANT): When considering Statement 1, the use of information from Statement 2 is prohibited. When considering Statement 2, the use of information from Statement 1 and the analysis derived from Statement 1 is prohibited. Both conditions can only be analyzed simultaneously during the stage where both Statements are considered together.
4. In sometime , both statement 1 and statement 2 can lead to answer alone.
5. Concluding with the Answer: End your response with "Answer: {answer}", where {answer} is the final result of your problem-solving process.

4+AutoForm:
Solve the problem presented below:
—
${task_description}
—

RESPONSE GUIDELINES:
1. Initial State Representation: Begin by providing a clear and detailed representation of the initial state or conditions of the problem.
2. Step-by-Step Solution Process:Progressively update the state representation as you work through each step of the solution. This should include all logical reasoning and calculations leading to the final answer.
3. To enhance clarity and eliminate ambiguities inherent in natural language, consider employing more structured and concise forms of communication for your step-by-step solutions. Suitable formats include code, pseudocode, JSON, markdown tables, logical operators, mathematical equations and so on.
4. You should consider three scenarios: using only Statement 1, using only Statement 2, and using both Statements.
5. Note (IMPORTANT): When considering Statement 1, the use of information from Statement 2 is prohibited. When considering Statement 2, the use of information from Statement 1 and the analysis derived from Statement 1 is prohibited. Both conditions can only be analyzed simultaneously during the stage where both Statements are considered together.
6. In sometime , both statement 1 and statement 2 can lead to answer alone.
7. Concluding with the Answer: End your response with "Answer: {answer}", where {answer} is the final result of your problem-solving process.

Table 13: prompt for Information Essentiality

PROMPT FOR HOTPOT QA, WIKI HOP, NARRATIVE QA

Shared portion of the prompt

You are ${agent_name}. Together with ${all_roles}, you are providing accurate answer to the user. Each of you will be provided parts of the contexts and a shared question.
EXAMPLE 1
—
# Context
${example_context_1}
# Question
${example_question_1}
—
${example_answer_1}.
EXAMPLE 2
—
# Context
${example_context_2}
# Question
${example_question_2}
—
${example_answer_2}.

Now the user gives you some contexts and the question:
—
# Context
${knowledge}
# Question
${task_description}
—

Baseline:

Given that each individual, including yourself, possesses unique contexts, it's essential to actively share and discuss this information with others to formulate a complete answer. Your specific context is unknown to others unless explicitly communicated. This collaborative effort is key to reaching an accurate answer based on the amalgamation of everyone's distinct contexts.

When you have reached the final answer, conclude it with "<A>xxx</A>", where "xxx" will be extracted and compared with ground truth. To end the conversation, all the players should end their responses with "<A>xxx</A>".
You are ${agent_name}. Now communicate with ${all_roles} to give the answer.

AutoForm(3.5-3.5,3.5-4,4-3.5,4-4):

Given that each individual, including yourself, possesses unique contexts, your specific context is unknown to others unless explicitly communicated.
You are ${agent_name}, collaborating with ${all_roles}, who are also intelligent assistants. Your goal is to provide a clear and concise answer to the user's question. Unlike typical communication, you will not use natural language, as it often contains ambiguities and emotional nuances. Instead, choose a more straightforward and precise communication medium, such as structured data, JSON, XML or code.
Now, start communicating with ${all_roles} using your selected non-natural language medium. Remember, clarity and brevity are key.
Once you have formulated the final answer,you must enclose it within "<A>xxx</A>", where "xxx" represents the answer phrase selected from the given choices. The conversation concludes when all participants have presented the same answer in this format.
If you have different opinion, explain it to your teammates.
Don't forget to enclose your answer within "<A>xxx</A>"

Table 14: prompt for Hotpot QA, Wiki Hop, Narrative QA