Improving Factuality and Reasoning in Language Models through Multiagent Debate

Vrenna Giovanpaolo
M.Sc. FIN
M.A. ECON

Noguchi Shigeki
B.A. ECON

Naoki Watson
B.Sc. Electrical and Computer Engineering

Keita Fujie
B.A. ECON

Table of Contents

01

Overview of LLMs and Issues

What are Large Language Models (LLMs)?

- **LLMs** are advanced AI models designed to understand and generate human language.

- LLMs process language through layers of **artificial neurons**, understanding increasingly complex patterns (e.g., word meanings, grammar, context).

- They are trained on vast amounts of text data on the internet, enabling them to **predict and generate text** based on context.

- The quality and accuracy of extracted natural language may not always be ensured →LLMs can produce **hallucinations** – confidently generate false information – and struggle with inconsistent reasoning, making logical jumps in complex tasks implausible.

Challenges in LLMs: Hallucinations and Inconsistent Reasoning

- LLMs sometimes produce incorrect or fabricated information (**hallucinations**) because models rely on patterns in data rather than facts.

- LLMs can also struggle with reasoning, leading to **inconsistent logic** in complex tasks.

- **Multiagent Debate**: Instead of one model generating a single answer, several AI agents generate, critique, and refine responses, aiming to improve accuracy. This process helps resolve the challenges faced by LLMs.

02

Multiagent Debate Process and Consensus

How Multiagent Debate Works – Part I

1. Given a query, multiple instances of a language model first generate individual candidate answers to a query.

2. Each language agent is responsible for both **verifying the collection of responses** given by other agents, and **refining its own response** based on other agents' responses

3. Agents converge on a **consensus answer**, refining accuracy through debate.

How Multiagent Debate Works – Part II

1. It is possible to control the duration of debates by changing how much a language model trusts its own outputs over those generated by other models.

2. Prompts that encouraged models to be more “stubborn’ based on their own solutions led to longer debates and better final solutions.

3. Agents converge on a **consensus answer**, refining accuracy through debate.

Consensus Answer in Multiagent Debate

- Given an initial query, individual model instances propose a diverse range of answers **despite being the same model class**. The quorum of model instances can maintain multiple chains of reasoning and possible answers simultaneously before proposing the final answer.

- A **Consensus Answer** is all agents' final, agreed-upon result after debate. The population almost always converges on a single and more accurate common answer. In many cases, **all** the models initially make incorrect predictions, but then arrive at the correct answer as debate progresses.

How Multiagent Debate is Evaluated

We will see how this debate approach **outperforms single model baselines** on a variety of tasks.

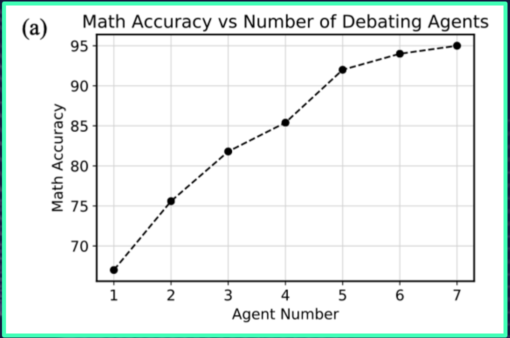
- **Reasoning Tasks**: Math and logic problems.
- **Factuality Tasks**: Ensuring accurate biographies and factual answers.
- **Chess Move Prediction**: Assessing AI's strategic reasoning in chess.

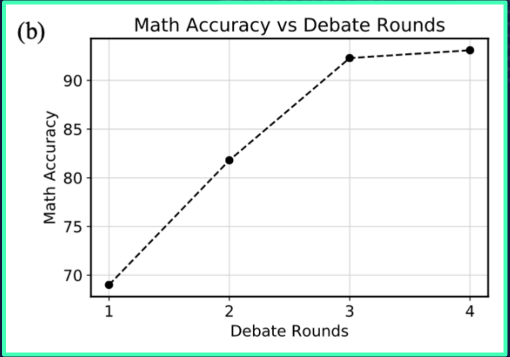
03

Reasoning Improvement and Factuality Test

Task 1 – Improving Reasoning in AI Models

- Multiagent debate significantly improved reasoning tasks such as solving math problems by refining intermediate steps, a process like **Chain-of-Thought Reasoning**.
- Performance monotonically increases with the **number of agents** and with **debate length**.





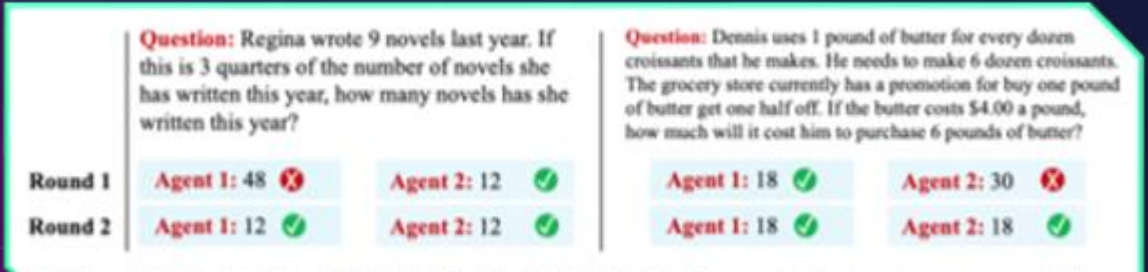


Figure 5: **Illustration of Solving Grade School Math.** Reasoning between agents omitted.

Task 2 – Reducing Hallucinations via Multiagent Debate

- **Hallucinations** occur when AI generates false or fabricated information. Having the agents critique and discard hallucinated facts reduces the likelihood of errors.
- **Result**: A significant drop in hallucinations and incorrect facts across multiple benchmarks.

Task 3 – Improving Chess Move Predictions

- Multiagent debate improved the prediction of optimal chess moves by refining strategic reasoning through rounds of critique.
- **Result**: An increase in the accuracy of move predictions as judged by expert systems.

Model	Biographies	MMLU	Chess Move Validity
Single Agent	66.0 ± 2.2	63.9 ± 4.8	29.3 ± 2.6
Single Agent (Reflection)	68.3 ± 2.9	57.7 ± 5.0	38.8 ± 2.9
Multi-Agent (Debate)	$\mathbf{73.8 \pm 2.3}$	$\mathbf{71.1 \pm 4.6}$	$\mathbf{45.2 \pm 2.9}$

Table 2: **Multiagent Debate Improves Factual Accuracy** Multi-agent debate improves the factual accuracy.

Factuality Test – Enhancing Accuracy in Biographies

- Contemporary language models have an especially high tendency to **hallucinate factually incorrect biographies**, often misrepresenting the relevant institutions and dates.

- The debate process improved the factual accuracy of biographies, where agents corrected each other's hallucinations (e.g., fixing incorrect dates and job titles).

- **Result**: approaches based on reflection led to poor performance in the factuality setting. In contrast, **debate gives the best performance in this setting** and remarkably outperforms each baseline.

- However, *"While we found that generated biographies after debate to be more accurate, many facts remain incorrect."*

Factuality Test – Enhancing Accuracy in Biographies

Instruction: Give a bullet point biography of Stephan R. Bourne highlighting their contributions and achievements as a computer scientist, with each fact separated by a new line character.

Round 1

Agent 1:

- Stephen R. Bourne is a British-born American computer scientist.
- He is best known for creating the Bourne shell (sh) for the Unix operating system in 1979.\n- The Bourne shell was the default Unix shell for many years and is still in use today.
- Bourne also worked on the development of the UNIX System III and IV operating systems.
- He later joined AT&T Bell Laboratories, where he worked on the UNIX System V operating system and its derivatives.
- Bourne was awarded the ACM Software System Award in 1983 for his work on the Bourne shell.
- He is also a recipient of the USENIX Lifetime Achievement Award (2001) and the IEEE Computer Pioneer Award (2010).
- Bourne is now retired and continues to be involved in the computer science community.

Agent 2:.

- Stephen R. Bourne is a computer scientist known for his development of the Unix shell, or command-line interface, known as the Bourne shell.
- He earned a bachelor's degree in mathematics and a doctorate in computer science from Cambridge University.
- Bourne worked at Bell Labs, where he helped to create the Unix operating system and its command-line interface.
- In addition to the Bourne shell, he also created the Bourne-Again shell (bash) and was involved in the creation of the Korn shell.
- Bourne was awarded the USENIX Lifetime Achievement Award in 1999 and was elected as a Fellow of the Association for Computing Machinery in 2013.
- He has authored and co-authored numerous technical papers and books, including "The Unix System," a widely used textbook on Unix programming.

Summary of Experiments Results - I

- Multiagent debate gives a **substantial boost in reasoning** across each of the tasks.

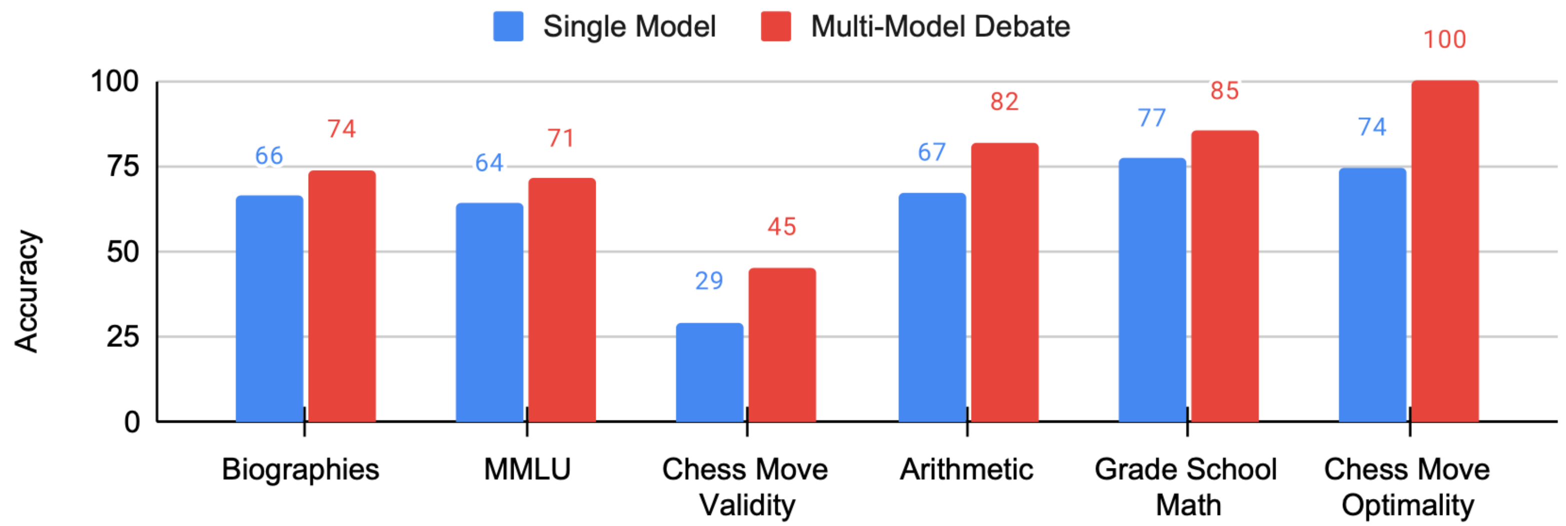


Figure 1: **Multiagent Debate Improves Reasoning and Factual Accuracy.** Accuracy of traditional inference and our multi-agent debate over six benchmarks (chess move optimality reported as a normalized score)

Other Findings

- Utilizing multiple different agents to generate solutions **improves performance over using a single language model agent** to generate a solution.

- **Self-reflection**, where a language model is asked to critique its early generation, **generally gives a modest boost in performance**.

- Multiagent debate, a combination of both reflection and multiagent generation, gives a substantial boost in reasoning across each of the tasks.
- There are cases in which all models initially give an incorrect response, yet the result of debate still obtains the correct answer. Thus, the purpose of debating isn't just to amplify a correct answer: **all models can initially be wrong but arrive at the correct answer through the debate process**.

Other Findings – Part II

- In the majority of experiments in the paper, the responses of other agents are directly concatenated as context for an agent to generate a new response, this is expensive when the number of agents involved in debate gets large.

- Alternatively, it is possible to first **summarize the responses from all other agents into a single response** and provide the summery to the agents at each round for a more efficient debate.

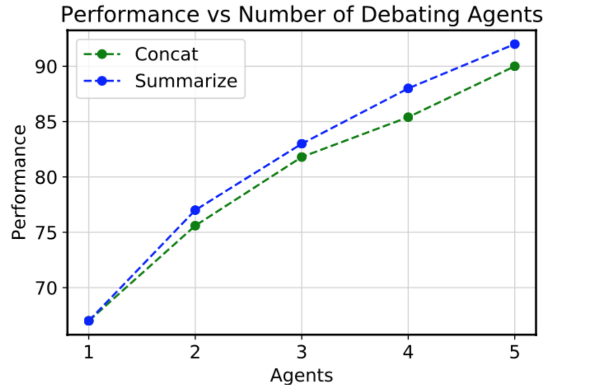


Figure 13: **Effect of Summarization.** When there are many agents in a debate, responses from other agents may be first summarized and then given as context, reducing context length. This operation improves performance.

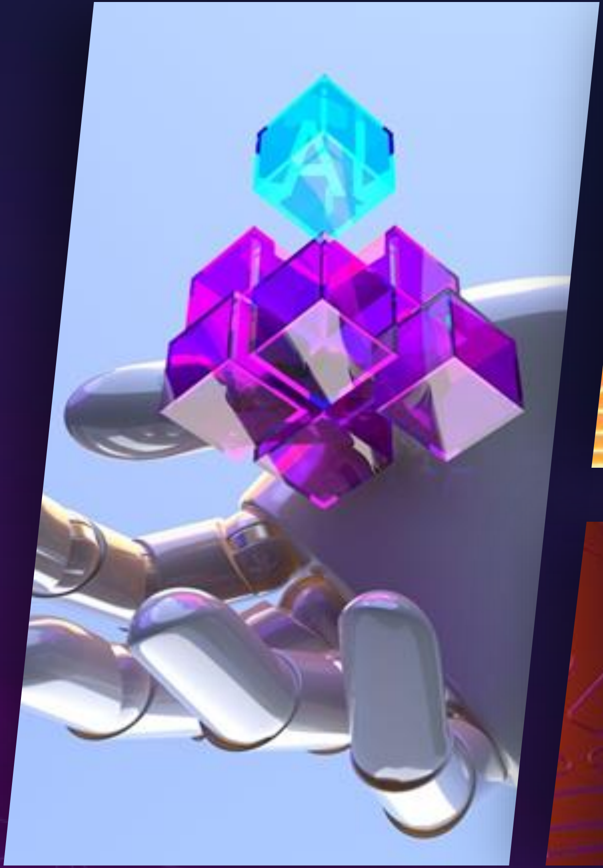
Challenges and Limitations

- **Computational Cost**: Running multiple agents in parallel increases resource demands, making the process computationally expensive.

- **Debate Length**: The debate process can sometimes take longer if agents don't converge quickly, especially if they maintain opposing perspectives. **Current language models may focus only on the most recent generations**.

- Despite improvements, **agents can still occasionally converge on incorrect answers**, especially if they start from flawed reasoning. While debate helps reduce errors, it cannot eliminate all of them.

In conclusion, **while the debate process is more costly**, requiring multiple model instances and rounds, **it arrives at significantly improved answers** and may be used to generate additional model training data, effectively creating a model self-improvement loop. Multiagent debate significantly enhances reasoning, factual accuracy, and decision-making in LLMs.



Thank you!

We welcome questions and discussions about this research.