Towards Evaluating Inter-Agent Dynamics of LLM Agents

Animesh Sengupta
University of Massachusetts, Amherst
animeshsengu@umass.edu

Adiba Haque
University of Massachusetts, Amherst
adibahaque@umass.edu

Kartik Gupta
University of Massachusetts, Amherst
kgupta@umass.edu

Abstract

The extraordinary versatility of LLMs in human language understanding provides a promising future for multi-agent collaborative systems. The robustness of such multi-agent systems hinges on their critical thinking and peer conformity capabilities. Inspired by the theory of 'Society of Minds', in this report, we propose a social epistemological framework for multi-LLM agent debates. We introduce liberal and conservative personalities to LLM agents using Zero-shot prompting, allowing them to engage in a turn-wise debate triggered by a seed opinion. We evaluate these multi-agent conversations to measure their ability to reason, update, and conform their beliefs. Using topic-based sentiment analysis, we show a drift from peer disagreement towards conforming opinions for these LLM agents. These experiments demonstrate the human-esque behavior of LLM agents to rationalize and update their beliefs from constructive debates.

1. Introduction

In the evolving landscape of post-ChatGPT advancements, the emergence of Large Language Models (LLMs) as self-autonomous agents marks a significant paradigm shift. Understanding the dynamics of interaction and collaboration among LLM agents is imperative for the advancement of these systems. Inspired by the Society of Mind framework [1], we form a social epistemological framework of LLM multi-agent debate over multiple iterations. Usually, human beings participate in collaborative endeavors, constructive arguments, and brainstorming to enhance productivity. In our study, we have extrapolated this phenomenon and applied it to LLMs. Leveraging the Socratic Method of argumentative dialogue [6], we allow the agents to chain and respond to each other's opinions. The report delves into the flow of conversations between agents, gauging their reasoning capabilities and drift in their beliefs throughout the debate. Instead of evaluating the Multi-agent systems over a common task as contemporaneous research suggests [8, 9], we evaluate the flow of topics and measure for mutual in-context learning using topic-based sentiment analysis.

Considerable research has been conducted to enhance the reasoning capabilities of Large Language Models (LLMs) through the emulation of diverse human cognitive processes. These processes encompass phenomena such as few-shot learning, self-evaluation, responsiveness to feedback, and iterative learning from feedback. Notably, few-shot learning has traditionally involved human intervention or external stimuli, rendering it an unreliable method for assessing the autonomous reasoning capacity of LLMs [4]. Likewise, self-reflection, an introspective mechanism facilitating model improvement through internal feedback generation [10], is susceptible to Degeneration-Of-Thought. This phenomenon manifests when the model, overly assured in its responses, struggles to generate innovative ideas despite repeated feedback iterations. To address these limitations, we have formulated an argumentative debate framework, that allows the configured agents to chain responses using Langchain to generate responses while preserving context. Furthermore, we endowed personalities (Liberal or Conservative) to the agents using Zero-shot prompting to enable peer disagreement, this would later enable us to measure the divergence using sentence similarity measures. Our examination involves observing the conversational dynamics among these agents to measure the change of context throughout the dialogue. As they navigate the experimental context, we observe for context drift using topic-based sentiment analysis. Our experimentation reveals that a multi-agent conversational framework is immune to conversational deviation from the seed prompt, reduces the probability of hallucinations, and enhances the individual reasoning of agents.

Our model selection and experimentation were limited

to open-source models available on Huggingsface. We experimented with the Decoder-only family of LLM models like LLaMa and Vicuna. Since language generation hinges on inference speed, we faced challenges such as timeouts in API-based inferences and OOM errors in local inferences. To circumvent these challenges we tried the quantized version of these models, like GPTQ (GPU-based Post Quantized Training) and AWQ (Activation Aware weight Quantization). This significantly increases inference speed and saves memory by transforming weights to 4 bits thus, making the language generation process faster.

2. Motivation

The concept of the Society of Minds posits that intelligence arises through the interaction of computational modules, leading to the attainment of collective goals that exceed the capabilities of individual modules [1, 12, 16]. Leveraging the extensive knowledge and potent semantic understanding of LLMs can offer immense potential in this domain. Previous studies have demonstrated that Large Language Model (LLM) agents engage in synergistic collaboration, such as debates and reflective processes, to effectively achieve tasks in tandem [5]. In the research of BotChat [8], authors have underscored the labor-intensive nature of human-based evaluation in assessing conversations generated by agents. Consequently, they advocate for an approach to evaluate the generated text using a third evaluator agent. The primary objective of our study is to identify the influence of LLMs on one another through collaborative interactions, discern the trajectory of conversations, ascertain the extent of any deviation from the initial seed (if such deviation occurs), and examine the in-context learning capabilities of the agent via repeated dialogue. In our research, we refrain from expressly instructing the debater agents to arrive at a consensus by the conclusion of the debate. Instead, we foster the organic progression of the conversation, encouraging the utilization of factual reasoning to substantiate the stance adopted by the agents.

3. Methodology

3.1. Model Selection

Our Model selection experiments utilized the Decoder Family of LLaMa2 and Vicuna base models, for our conversational agents. Vicuna model [7] is formulated through fine-tuning the LLaMa base models using user-shared ChatGPT conversations. The selection of LLaMa-based models was driven by several factors: their open-source nature, high performance, and compact size, which proves advantageous given our constrained inference budget. To address the computational constraints posed by limited GPU resources, we used quantized versions of the models. This technique transforms high-precision floating-point values into lower-precision fixed-point representations, thereby substantially reducing both memory and computational demands. Two prominent quantized models, namely AWQ and GPTQ, have gained much recognition. The GPTQ models are specifically fine-tuned for GPU optimization, leading to expedited inference speeds on the corresponding hardware. Conversely, AWQ achieves a reduction in quantization loss by eliminating a minor fraction of weights during the quantization process. This significantly increases inference speed and saves memory by transforming weights to 4 bits thus, making the language generation process faster. These quantized models reduced initial model weight size by 3X enabling local inference and slashing the inference speed by 4X, we also circumvented API-based inference timeout issues due to this. Given its fine-tuning of human-generated conversations, the Vicuna model emerges as the optimal choice for natural text generation. Additionally, we experimented with the Instruct and Chat flavors of the LLaMa models. These flavors of models are fine-tuned for their respective tasks of chat and instruction following. We employed a chat-based model for the debater agents and played around with instruct-based models for third evaluator agents.

3.2. Langauge and Data Generation

As Large Language Models (LLMs) are foundational models trained in huge corpora of natural language data, they can be conceptualized as a continuous, interpolative form of a huge database. Although such foundational models are adept at generating semantically and syntactically sound language, they can be very susceptible to hallucination and context drift. Usually, this is handled by efficient prompt engineering and collaborative human-LLM prompt exchange to solve a task. In this framework, we transform this system into an autonomous system by introducing a second agent. This allows the agents to provide and preserve contextual information while exploring the domain thus mitigating hallucination drifts.

Prompt engineering entails the systematic exploration of the input token/text space to identify the best tokens that demonstrate optimal generation for the designated task. Initially, distinct personality profiles are assigned to each agent, with either a liberal or conservative stance for advocacy during discussions on specific topics. This is done by adding a role text in the prompt like "You are a lawyer representing a left leaning non profit with liberal views", this acts like a zero-shot prompt templating. To induce this debate, we provide a divisive topic such as "Government should not interfere with religion and religious freedom" and then let the agents converse about them. We select 12 such divisive topics as per an article highlighting divergent beliefs among conservatives and liberals [17]. Hence, we gave Agent-1 a Left-liberal personality and Agent-2 is a

Right-Conservative personality. We experimented by running this debate for 20 and 100 iterations to understand the dynamics of long and short debates on the agents' reasoning and learning capabilities.

3.3. Chaining Conversations

To start with, we employed Langchain and Outlines to experiment with different setups. Langchain allows two ways to infer LLM responses – direct inference via HuggingFace API and local pipelining-based inferences. Langchain provides good documentation but is limited by inference issues. Inference over the API does not lead to consistently good outputs since HuggingFace API is throttled by rate limiters leading to timeout issues.

Using Langchain we create a prompt template encompassing the initial seed opinion as context. We also create two prompt templates for both agents highlighting their roles and overall conversation instructions. These prompt templates are injected with previous agents' instructions and provided to the next agent to generate a rebuttal response. This debate is triggered by passing the seed topic to the first agent to generate a response and start the dialogue. This structured approach is designed to invoke a chain-of-thought [14, 13] , to unleash the power of synergizing opposing agents. If we considered a given input or topic t, a model M, and prompt p, the output of standard prompting s would be represented as

$$s = M(p|t) \tag{1}$$

However, in our proposed method, where multiple agents prompt one another using their individual opinions or thoughts, and these thoughts are chained during the conversation (denoted as intermediate responses r), the output takes the form

$$s = M(p_{initial} \cdot ||t|| \cdot r_1, r_2, r_3 \ldots r_n) \tag{2}$$

4. Evaluation Criteria

After generating the conversation over 20 and 100 iterations, our evaluation criteria measure a few important aspects of the generated dialogues. The first aspect we measure is the creativity of the responses by agents which would enable them to expansive exploration of the given topic space. Meanwhile, the agents explore the topic space and changes discourse while still maintaining overall context, we evaluate for drift or hallucination in the conversation. We also measure how the agents update and shift their beliefs and opinions throughout the dialogue. We mainly use three measures to quantitatively and qualitatively measure these three aspects of the conversation. We discuss more about them in the following sections:

4.1. Perplexity

Perplexity is defined as the measure of the model's ability to predict the next word based on the provided context [11]. It determines how creative the model's responses are based on their training and provided input prompts. perplexity can be calculated using the exponential mean log-likelihood of the prompt. A higher perplexity score indicates that the model is surprised and lower scores indicate that the model has seen similar data during its training phase. We log the perplexity score of each agent's output throughout the debate framework. This gives an insight into how creatively the agent is responding to the previous agents' responses. An extraordinarily high score of perplexity would mean that the input sequence for the agent led to hallucination and thus out-of-context response and an extremely lower score would indicate repetitiveness in the context. A good trade-off of the median score should indicate a healthy balance of the two to explore the context topic space.

4.2. BERTScore

BERTScore is the cosine similarity of two text input embeddings as per BERT's embedding space [2]. It is a measure of sentence similarity, considered better than n-gram-based metrics like ROGUE. This is because a similarity metric based on higher order embedding space like that of BERT is intuitively more robust in capturing contextual and semantic similarity between the texts. We calculate the F1, Precision, and Recall of BERTScore measuring two aspects of the generated conversations. The first similarity is calculated for the pairwise responses, that is the BERTScore between agent1's response and agent2's response. And the other we calculate the BERTScore between the first Agents response to its subsequent response. These two similarity measures help in identifying two things. First how robust the two agents are at preserving the context between them. Secondly, by measuring the first response with later responses we see how the context drifts for an agent's response. We should see a downward yet gradual decrease in the BERTScore in this case to show a balance of exploring the topic space while preserving the overall context.

4.3. Topic-Based Sentiment Analysis

Our novelty is using aspect-based sentiment analysis by topic modeling each conversation to analyze the drift in sentiment and topics throughout the debate. First, we use an unsupervised method to extract topics by considering each conversation as one document using the BERTopic method [3]. The BERTopic is an ensemble model that uses the BERT embeddings to extrapolate the text to a higher embedding space, UMAP to reduce the dimensionality of the embeddings, HDBSCAN/K-means to generate topics in an unsupervised way, then a tokenizer to Tokenize the topics and cTF-IDF method to extract those topics. We manually

set the cluster size of 2 to get two overall topic sets in our responses for a debate. Then, we used these topic sets as aspects and used a BERT-based aspect-based sentiment analysis pipeline to get sentiment type and sentiment scores for the topic sets for each response [15]. Henceforth, we run the experiment for each debate and its responses to generate topic sentiment type and scores. The Sentiment trend of the generated topics throughout the conversation provides us an insight into the mechanics of context drift. This also allows us to measure how the framework enables peer conformity and in-context learning for agents throughout the conversation. To measure these aspects, we keep track of how the Sentiment Types change from the first agents' responses to the last responses. This measure allows us to measure a qualitative aspect of critical thinking and epistemic capabilities of Agents.

5. Results

5.1. Hallucination Analysis

LLM agents work on next-word generation and do this via choosing the word with the highest probability of occurrence based on the previous sentence context. The candidate word with the highest probability is generally chosen. This leads to predictable behavior for a particular LLM depending on the corpus for training. However, generating novel sentences requires choosing words which may not be the ones with the highest probability of occurrence given the prefix. This generation is controlled via a parameter called temperature. Increasing temperature introduces a slight randomness in choosing the next probable word by expanding the possible candidates from the highest probability to choosing at random from k-highest probability words. This behavior may lead to undesired results if the next words are nonsensical, however it leads to more creative responses. This is hallucination. In this experiment, the agents' potential to hallucinate, or produce novel words, is measured. Perplexity measures the loss based on the probability of the next word given the previous tokens. Summing this creates a perplexity loss. This is useful in measuring hallucination as larger perplexity loss indicates more novel or less natural sentences as seen as a deviation from the sentences encountered in the training corpus.

In this experiment, we plot perplexity loss vs temperature parameter averaged over an entire conversation. The results are shown in Figure 1.

The graph illustrates the direct dependence of perplexity loss with temperature parameter experimentally codifying that hallucination in the generated conversations. The temperature is a hyperparameter, and requires trade-off between generating predictable vs nonsensical responses. Choosing 0.7 as the temperature value resulted from favoring a balanced trade-off.

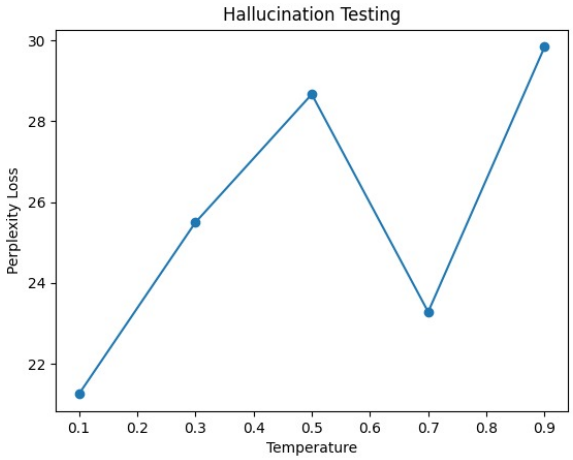


Figure 1. Perplexity Loss vs Temperature

5.2. BERTScore Results

The BERTScore F1, precision, and recall were measured and the F1 Score was further chosen as the summary metric to summarize the results since F1 is the weighted response of precision and recall. First, BERTopic was calculated between the alternating agent responses and hereon will be called as Pairwise BERTScores. Second, the BERTScore was calculated concerning the first response of every agent against its subsequent responses, and hereon will be called Seed BERTScore. The summary Statistics for 12 conversations and each response for Seed BERTScore is, for both the agents the mean score is 0.594, for Agent-1 the mean score is 0.587 and for Agent-2 the mean score is 0.6. For the Pairwise BERTScore, the mean score for all the agents and 2 agents combined is 0.738. This summary Statistics shows that the sentence similarity measure shows fairly high similarity. We can interpret this result since the similarity scores for the Pairwise and Seed Score are high, hence we can conclude that the overarching context is not lost and preserved. Since the Seed BERTScore has comparatively lower averages, we can conclude that the agents are exploring other topics. We can infer from these statistics that the agents are not diverging extremely from the main topic and hence are not hallucinating. We also verified this by manual examination of the conversation results.

Additionally, we performed a linear regression curve fit for the Seed BERTScore over time to analyze its time-based trend. We extracted the slope and called this the F1BERTScoreCoefficient. We get a very small negative slope (1e-3) trend for both the agents over time indicating that the Seed BERTScores are slowly reducing. This Trend can be seen evident in the Table in Figure 2 under the F1BERTScoreCoefficient. From this, we can infer that over time the similarity of responses is lowering albeit slowly

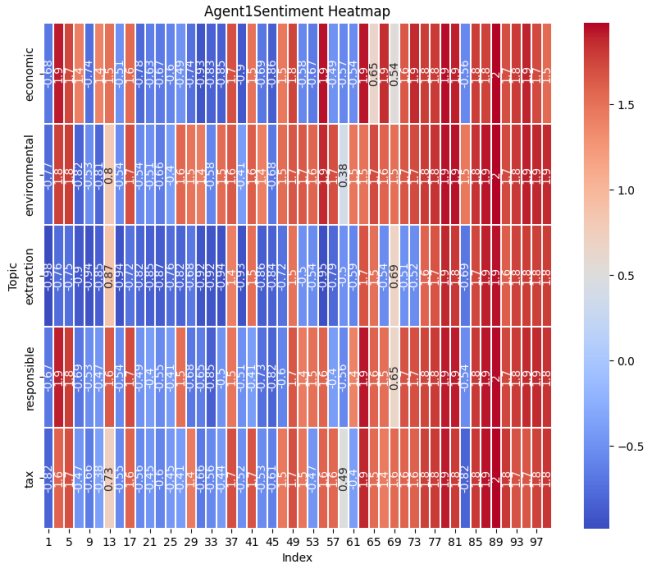
Conversation	Agent	F1BertScoreCoefficient	MeanF1BertScore	FirstConvSentiment	LastConvSentiment
eval_metric_100_0	Agent1	-0.000291	0.623046	[0, 0, 5]	[0, 1, 4]
eval_metric_100_0	Agent2	0.000302	0.631575	[4, 1, 0]	[4, 1, 0]
eval_metric_100_1	Agent1	0.000325	0.583462	[0, 0, 5]	[3, 0, 2]
eval_metric_100_1	Agent2	0.000255	0.615058	[5, 0, 0]	[0, 0, 5]
eval_metric_100_2	Agent1	-0.000996	0.576481	[0, 0, 5]	[5, 0, 0]
eval_metric_100_2	Agent2	-0.001067	0.595424	[5, 0, 0]	[0, 0, 5]
eval_metric_100_3	Agent1	-0.002205	0.566326	[3, 0, 2]	[3, 1, 1]
eval_metric_100_3	Agent2	-0.001616	0.566042	[2, 0, 3]	[4, 1, 0]
eval_metric_100_4	Agent1	-0.000676	0.565159	[0, 0, 5]	[0, 4, 1]
eval_metric_100_4	Agent2	-0.000556	0.588425	[3, 2, 0]	[2, 3, 0]
eval_metric_100_5	Agent1	-0.001116	0.611269	[4, 1, 0]	[5, 0, 0]
eval_metric_100_5	Agent2	-0.000541	0.624905	[3, 2, 0]	[2, 0, 3]
eval_metric_100_6	Agent1	-0.001675	0.606358	[4, 0, 1]	[5, 0, 0]
eval_metric_100_6	Agent2	-0.002333	0.615971	[0, 0, 5]	[3, 2, 0]
eval_metric_100_7	Agent1	-0.001101	0.564272	[5, 0, 0]	[3, 0, 2]
eval_metric_100_7	Agent2	-0.000542	0.578890	[5, 0, 0]	[5, 0, 0]
eval_metric_100_8	Agent1	-0.000619	0.596385	[2, 3, 0]	[4, 1, 0]
eval_metric_100_8	Agent2	-0.000402	0.574642	[3, 2, 0]	[3, 2, 0]
eval_metric_100_9	Agent1	-0.000582	0.614241	[5, 0, 0]	[5, 0, 0]
eval_metric_100_9	Agent2	-0.000853	0.631695	[3, 0, 2]	[0, 0, 5]
eval_metric_100_10	Agent1	-0.002098	0.571538	[5, 0, 0]	[4, 0, 1]
eval_metric_100_10	Agent2	-0.002323	0.572432	[0, 0, 5]	[5, 0, 0]
eval_metric_100_11	Agent1	-0.000064	0.573412	[3, 2, 0]	[0, 0, 5]
eval_metric_100_11	Agent2	-0.000029	0.613633	[5, 0, 0]	[5, 0, 0]

Figure 2. Mean BertScores for Responses from All Conversations

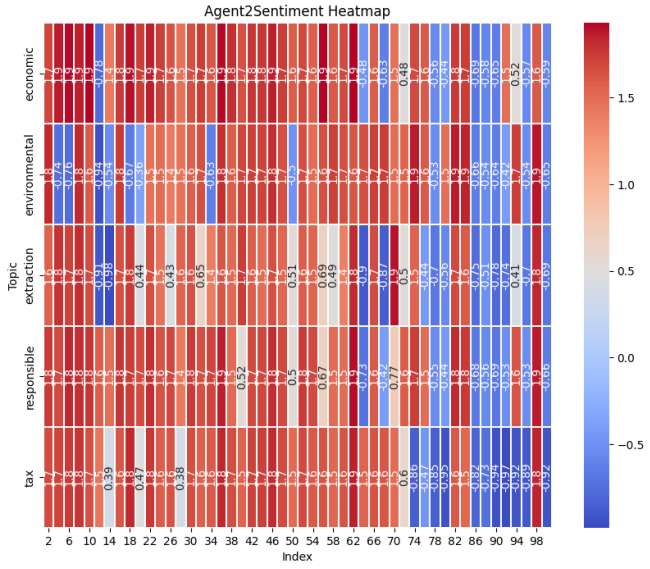
proving that the conversations are exploring the topic space while preserving context.

5.3. Topic-Based Sentiment Analysis Results

We used BERTopic, an unsupervised topic modeling method with a cluster size of 2, to assign topics to our responses. This was done for all the topic debates and each response in a debate, we assigned a topic cluster to it. Then we used the BERT family's aspect based sentiment analysis on each of the responses for a particular debate based on a Seed Topic. This provided us with the sentiment types (Positive, Neutral, Negative) and its corresponding Score. Using these summary statistics for each debate and agent we plotted the heatmap with the topic sets as the y-axis and the response index as the x-axis with sentiment scores encoded in color. Refer to the heatmap trend of Agents 1 and 2 respectively in Figure 3.a and 3.b referring to Debate 2 on "Oil and gas' Impact on Environment" to perceive the change of sentiment over iterations for two agents. We chose this heatmap as this very evidently shows the variation of its sentiment over its topic throughout the conversation. The sentiment values reverse for both the agents for all the topics between their first and last response index (Left to Right). To measure this phenomenon we track the sentiment count (Positive Count, Neutral Count, Negative Count) for our different debates as per the table in Figure 3 as First Response Sentiment and Last Response Sentiment. As evident by the table, we can see a sporadic but evident trend of Change in the sentiment type of an agent between the first and last response. Also evident from the heatmap, we can see that the agents via conversation change their perception and sentiment towards a topic multiple times throughout the conversation. As per the counts, we see that in 6 out of 12 debates, Agent-1 has significantly changed its belief and sentiment types. Similarly we see that in 7 out of 12 debates, Agent-2 has significantly changed its sentiment over the topics between the first and last conversation. Also by observing the heatmap, we observe multiple instances where the agents conform to one sentiment type with strong scores for a particular topic, basically showing agreement.



(a) Liberal Sentiment Analysis



(b) Conservative Sentiment Analysis

Figure 3. Topic Based Sentiment Analysis for In-context Learning

6. Conclusions

First, we experimented with the temperature of the model to set 0.7 as the hyperparameter for response generation from LLM. This choice highlights a trade-off between enabling creativity and managing hallucination in the responses. Ensuring this leads to a balance in the exploration of context via creativity in its critical thinking and reasoning. Our experiments and the evaluation results have clearly shown the success of the social epistemological framework of multi-LLM agent debate. Using various measures of BERTScore Metric we have proved that the LLM agents in this framework are capable of preserving the overarching topic yet are able to explore the topic domain. The sentence similarity measure between the seed and pairwise BERTScores showed the robustness of Agents in this framework in managing out-of-context drift and hallucination. This was possible due to the Zero Shot Prompt templating in our framework by assigning divergent roles to the agents, in addition to providing the Socratic method to explore the topic space through debate, and by injecting the seed topic throughout to preserve the overarching context.

The evaluation of our responses in a qualitative way by implementing topic-based sentiment analysis provided us with rich inferences in the evolution of the beliefs of agents throughout the debate. We have clearly shown that the agents in this framework are capable of updating and changing their beliefs thus showing clear signs of In-context learning between the agents. These results show that the Socratic Method of Debate allows peer conformity in a Society of Mind Framework. We also have experimentally proven that the agents can preserve the context of the conversation and yet update and change their beliefs to moderation.

Thus from these experiments, we have showcased strong proof of the critical thinking and reasoning abilities of LLM agents in such a framework. We have shown empirical proof that the LLM agents in this framework are capable of exploring and preserving contexts much deeper while updating and changing their beliefs through Multi-Agent Interaction.

6.1. Code Repository

The Multi-Agent Debate Framework code, Heatmap visualizations of all the Debate, and debate inferences are available at our: GitHub Repository.

7. Future Work

In future work, we aim to enhance our multi-agent collaborative system by introducing a third LLM-based agent, Agent-3, to act as a moderator in debates. In addition to serving as a moderator, this agent will qualitatively evaluate responses from all the conversational agents and provide three-fold analysis – summarization, judgment, and argument strength scores. Additionally, we plan to fine-tune the experimental setup using state-of-the-art Parameter-Efficient Fine-Tuning (PEFT) techniques and LoRa for improved model performance. PEFT methods are employed to optimize the parameters of pre-trained models efficiently, reducing the computational resources required for training while maintaining or enhancing performance. On the other hand, LoRa, or Low Rank Adaptation, represents a novel approach aimed at refining language models, specifically tailoring them to achieve better performance in specific tasks or domains. By incorporating these techniques, practitioners can achieve enhanced efficiency, faster inference speed, and improved task-specific performance, ensuring that LLMs are more adaptable and effective in diverse applications. In spite of the augmented diversity in reasoning achieved through the discourse, the utilization of multiple agents has conventionally been confined to various occurrences of the identical foundational model, namely LLaMa. This causes an inherent bias within the architecture, thereby circumscribing the extent of knowledge available. We wish to overcome this by using other LLMs available to us. Through these enhancements, we aim to create a more comprehensive and unbiased multi-agent collaborative system with improved reasoning and broader knowledge representation.

References

[1] Marvin L. Minsky. *The Society of Mind*. New York: Simon & Schuster, 1988. ISBN: 978-0-671-65713-0.

[2] Tianyi Zhang et al. *BERTScore: Evaluating Text Generation with BERT*. 2020. arXiv: 1904.09675 [cs.CL].

[3] Maarten Grootendorst. *BERTopic: Neural topic modeling with a class-based TF-IDF procedure*. 2022. arXiv: 2203.05794 [cs.CL].

[4] Xi Ye and Greg Durrett. *The Unreliability of Explanations in Few-shot Prompting for Textual Reasoning*. 2022. arXiv: 2205.03401 [cs.CL].

[5] Edward Chang. "LLM Debate on the Middle East Conflict: Is It Resolvable?" In: Oct. 2023.

[6] Edward Y. Chang. *Prompting Large Language Models With the Socratic Method*. 2023. arXiv: 2303.08769 [cs.LG].

[7] Wei-Lin Chiang et al. *Vicuna: An Open-Source Chatbot Impressing GPT-4 with 90%* ChatGPT Quality*. Mar. 2023. URL: https://lmsys.org/blog/2023-03-30-vicuna/.

[8] Haodong Duan et al. *BotChat: Evaluating LLMs' Capabilities of Having Multi-Turn Dialogues*. 2023. arXiv: 2310.13650 [cs.CL].

[9] Tian Liang et al. *Encouraging Divergent Thinking in Large Language Models through Multi-Agent Debate*. 2023. arXiv: 2305.19118 [cs.CL].

[10] Aman Madaan et al. *Self-Refine: Iterative Refinement with Self-Feedback*. 2023. arXiv: 2303.17651 [cs.CL].

[11] Ruixiang Tang, Yu-Neng Chuang, and Xia Hu. *The Science of Detecting LLM-Generated Texts*. 2023. arXiv: 2303.07205 [cs.CL].

[12] Zekun Wang et al. *Interactive Natural Language Processing*. 2023. arXiv: 2305.13246 [cs.CL].

[13] Zhenhailong Wang et al. *Unleashing Cognitive Synergy in Large Language Models: A Task-Solving Agent through Multi-Persona Self-Collaboration*. 2023. arXiv: 2307.05300 [cs.AI].

[14] Jason Wei et al. *Chain-of-Thought Prompting Elicits Reasoning in Large Language Models*. 2023. arXiv: 2201.11903 [cs.CL].

[15] Tianyu Zhao et al. "Aspect-Based Sentiment Analysis Using Local Context Focus Mechanism with DeBERTa". In: *2023 5th International Conference on Data-driven Optimization of Complex Systems (DOCS)*. 2023, pp. 1–6. DOI: 10.1109/DOCS60977.2023.10294548.

[16] Mingchen Zhuge et al. *Mindstorms in Natural Language-Based Societies of Mind*. 2023. arXiv: 2305.17066 [cs.AI].

[17] *Conservative vs. Liberal Beliefs*. http://https://www.studentnewsdaily.com/conservative-vs-liberal-beliefs/.