Contents lists available at ScienceDirect

AI Open

journal homepage: www.keaipublishing.com/en/journals/ai-open

Full length article

ChatLLM network: More brains, more intelligence

Rui Hao[a], Linmei Hu[b,*], Weijian Qi[c], Qingliu Wu[a], Yirui Zhang[a], Liqiang Nie[d]

[a] *School of Computer Science, Beijing University of Posts and Telecommunications, Beijing, 100876, China*
[b] *School of Computer Science, Beijing Institute of Technology, Beijing, 100081, China*
[c] *Department of Automation, Xi'an Jiaotong University, Xi'an, 710049, Shaanxi, China*
[d] *School of Computer Science and Technology, Harbin Institute of Technology, Shenzhen, 518055, Guangdong, China*

ARTICLE INFO

Keywords:
ChatGPT
Large language model

ABSTRACT

Dialogue-based language models mark a huge milestone in the field of artificial intelligence, by their impressive ability to interact with users, as well as a series of challenging tasks prompted by customized instructions. However, the prevalent large-scale dialogue-based language models like ChatGPT still have room for improvement, such as unstable responses to questions and the inability to think cooperatively like humans. Considering the ability of dialogue-based language models in conversation and their inherent randomness in thinking, we propose ChatLLM network that allows multiple dialogue-based language models to interact, provide feedback, and think together. We design a network of ChatLLMs, consisting multiple layers of language models. Specifically, individual instances of language model may possess distinct perspectives towards the same problem, and by consolidating these diverse viewpoints via a separate language model, the ChatLLM network system can conduct decision-making more objectively and comprehensively. In addition, a language-based feedback mechanism comparable to backpropagation is devised to update the outputs of the language models within the network. This stratified system of interaction can be analogized to the relationship between leaders and employees in a social organization, where collective decision-making often yields superior judgments or resolutions. Experiments on datasets demonstrate that our network attains significant improvements in problem-solving, leading to observable progress amongst each member.

1. Introduction

Large language models have attracted widespread attention in the field of artificial intelligence because of their impressive ability to solve natural language processing tasks. Dialogue-based large language models, such as ChatGPT, in particular, have exerted a significant impact on the development of society and have become an exemplar of artificial intelligence applied to daily life, attracting extensive attention from both academia and industry. It becomes extremely challenging to distinguish them from humans, solely based on their speech style and content.

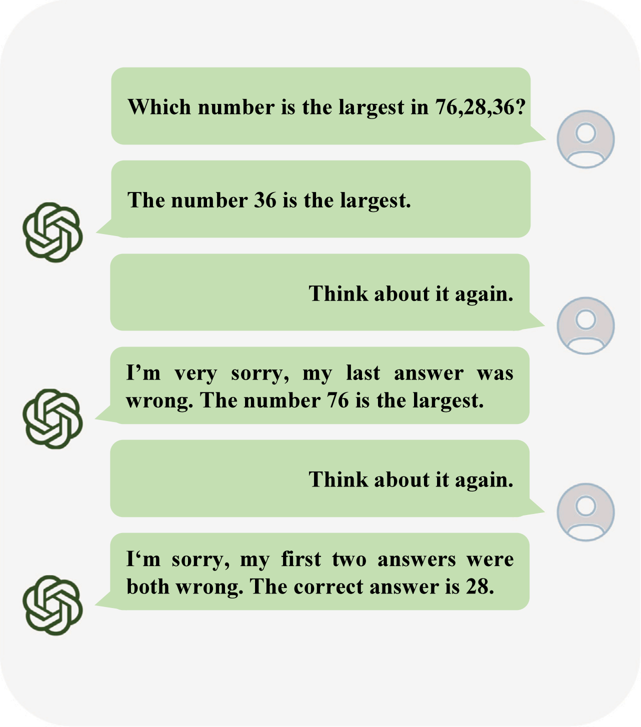
Despite their impressive capabilities in interacting with humans and handling various natural language processing tasks, dialogue-based large language models like ChatGPT may still provide unsatisfactory responses in certain conversational scenarios. This is because these models are based on generative models that rely on statistical patterns in the data they were trained on, rather than in-depth understanding or true comprehension of the content. We observe two distinct aspects of these unsatisfactory responses. The first aspect is instability, in which the answers can significantly vary despite the same context and prompt being provided, as shown in Fig. 1.

The second is incomprehensiveness, as a single instance of model may easily provide one-sided answers, failing to engage in collaborative thinking for more comprehensive answers. The model's responses show stochastic fluctuations around the fact when faced with challenging questions, so it is necessary to design an effective method to consolidate multiple model outputs for balancing such variance (Wei et al., 2022b).

In this work, to address the above potential issues with a single dialogue-based language model, we propose a multi-layer ChatLLM network model to aggregate the viewpoints of other models layer by layer, analogous to the social dynamics between leaders and employees, where making decisions hierarchically and collectively can ultimately enhance the overall performance. In particular, we first devise a forward aggregation mechanism where the leader ChatLLM at higher layer aggregates the lower-layer employee models' outputs. Subsequently, a language-based backpropagation mechanism is employed to learn

[*] Corresponding author.
E-mail addresses: haorui@bupt.edu.cn (R. Hao), hulinmei@bit.edu.cn (L. Hu), qiweijian@stu.xjtu.edu.cn (W. Qi), wuql@bupt.edu.cn (Q. Wu), zhangyirui@bupt.edu.cn (Y. Zhang), nieliqiang@gmail.com (L. Nie).

https://doi.org/10.1016/j.aiopen.2025.01.001
Received 28 November 2023; Received in revised form 6 December 2024; Accepted 2 January 2025
Available online 12 February 2025



Which number is the largest in 76,28,36?

The number 36 is the largest.

Think about it again.

I'm very sorry, my last answer was wrong. The number 76 is the largest.

Think about it again.

I'm sorry, my first two answers were both wrong. The correct answer is 28.

Fig. 1. An example of instability of ChatGPT.

from their mistakes and improve their performance over time by incorporating feedback and updating their thinking processes. Moreover, the dropout mechanism is introduced to regularize the inputs for each ChatLLM model and thus prevent overfitting.

In addition, we do not necessarily require the models to use ChatGPT in the proposed network, as the entire network may become stronger with the enhancement of ChatLLMs or with the use of different types of ChatLLMs. The main contributions of this article can be summarized as follows:

(1) We propose a novel multi-layer ChatLLM network which enables multiple dialogue-based language models to interact, and hence enhancing problem-solving abilities.
(2) In the proposed ChatLLM network, we design a forward aggregation mechanism to consolidate the outputs. We also putforward a novel language-based backpropagation algorithm to update the network.
(3) Experiments show significant improvements compared to the vanilla ChatLLM model and simple ensemble model of ChatLLMs. As a fundamental research, our study could provide valuable insights and inspirations for synthesizing multiple ChatLLMs in future work.

2. Related work

2.1. Large language models

The introduction of the transformer model (Vaswani et al., 2017) has made it possible to train large-scale unsupervised text data. In the past few years, encoder-based models such as BERT (Devlin et al., 2019) have demonstrated impressive capabilities in various natural language processing (NLP) tasks. More recently, decoder-based models such as GPT-1 (Radford and Narasimhan, 2018), GPT-2 (Radford et al., 2019), and T5 (Raffel et al., 2020) have made even greater strides. As the number of model parameters has increased, models like GPT-3 (Brown et al., 2020), often referred to as large language models, have gradually acquired zero-shot learning abilities, which have the capacity to generate responses based on instructions without requiring any examples. ChatGPT, also known as InstructGPT, is an advanced version of the GPT-3 model, enhanced by instruction tuning (Wei et al., 2022a), and reinforcement learning from human feedback (RLHF) (Bradley Knox and Stone, 2008) (Ouyang et al., 2022). Unlike the original GPT-3 models, the InstructGPT models, after fine-tuning for user instructions, demonstrate a considerably enhanced capability to generate more aligned and helpful outputs in response to user instructions.

2.2. Collaboration for LLM-based agents

In recent years, systems where various Large Language Model (LLM) agents collaborate have exhibited remarkable performance in a multitude of tasks. Within such systems, each agent is actively involved in exchanging information via natural language (Guo et al., 2024; Xi et al., 2023).

Cooperative multi-agent systems have gained extensive application. A notable example is CAMEL (Li et al., 2023), which exemplifies a successful dual-agent cooperative system. Within a role-playing communication framework, agents assume the roles of AI Users and AI Assistants. Furthermore, the AgentVerse (Chen et al., 2024) develops a comprehensive and multi-task-tested framework that facilitates the cooperation of group agents. It enables the assembly of an agent team that can dynamically adjust to the complexity of the task at hand. Meanwhile, MetaGPT (Hong et al., 2024) takes its cue from the classic waterfall model in software development, formalizing the inputs and outputs of agents as standardized engineering documents.

Additionally, it is increasingly acknowledged by researchers that within LLM-based multi-agent systems, the emergence of beneficial changes among agents can spontaneously arise through mechanisms such as competition, argumentation, and debate (Irving et al., 2018). SocraSynth (Chang, 2024) provides foundational insights into managing biases and eliminating hallucinations through contentious debates among LLMs. In reasoning tasks, Du et al. (2024) propose the concept of debate, allowing agents to incorporate feedback from their peers. When these responses differ from the agent's own judgments, an argumentation process occurs, leading to more refined solutions. Using role-playing, ChatEval (Chan et al., 2024) establishes a multi-agent team of referees. Through self-initiated debates, these agents evaluate the quality of the text produced by LLMs, achieving a standard comparable to human evaluators.

2.3. Improving language models via feedback

Recently large language models (LLMs) have shown great potential in improving their performance and generating high-quality text by incorporating iterative feedback mechanisms. Madaan et al. (2023) proposed SELF-REFINE, a network that leverages iterative feedback and refinement to improve initial outputs from LLMs. The approach allows a single LLM to generate an output, provide multi-aspect feedback on its own output, and refine it based on the feedback, leading to better results across a range of tasks. Press et al. (2023) investigated the compositionality gap in GPT-3 models and presented the self-ask method to enhance compositional reasoning. SelfCheck (Miao et al., 2024) is a system that enables agents to assess and correct their reasoning at different points in the process. InterAct (Chen and Chang, 2023), on the other hand, utilizes various language models in supporting roles to assist the primary model in avoiding errors and inefficiencies. Reflexion (Shinn et al., 2023; Wang et al., 2024) is designed to boost an agent's planning capabilities through comprehensive verbal feedback. In this model, the agent initially takes an action that is informed by its memory, followed by the evaluator providing feedback based on the agent's trajectory. Additionally, ChatCoT (Chen et al., 2023) leverages feedback from an evaluation module that records the agent reasoning steps to refine its reasoning capabilities.

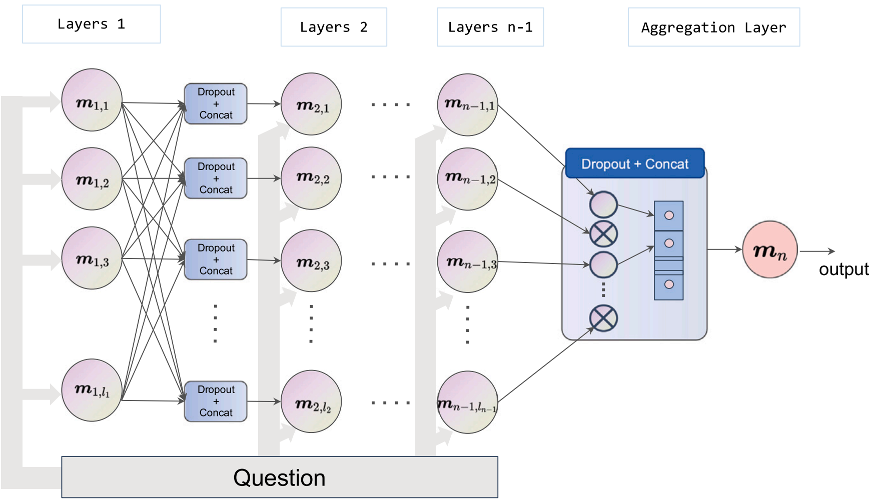
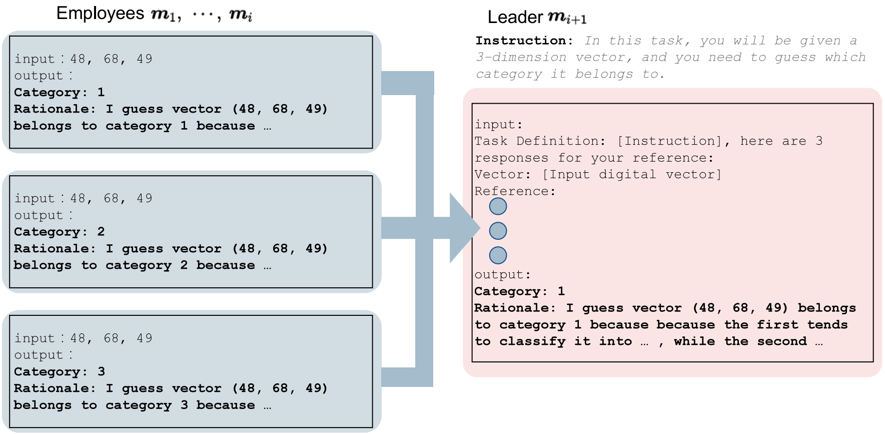


Fig. 2. Network architecture and forward process.



Employees $m_1, \cdots, m_i$

```
input：48, 68, 49
output：
Category: 1
Rationale: I guess vector (48, 68, 49)
belongs to category 1 because …
```

```
input：48, 68, 49
output：
Category: 2
Rationale: I guess vector (48, 68, 49)
belongs to category 2 because …
```

```
input：48, 68, 49
output：
Category: 3
Rationale: I guess vector (48, 68, 49)
belongs to category 3 because …
```

Leader m_{i+1}

```
Instruction: In this task, you will be given a
3-dimension vector, and you need to guess which
category it belongs to.
```

```
input:
Task Definition: [Instruction], here are 3
responses for your reference:
Vector: [Input digital vector]
Reference:
○
○
○
output:
Category: 1
Rationale: I guess vector (48, 68, 49) belongs
to category 1 because because the first tends
to classify it into … , while the second …
```

Fig. 3. Illustration of the forward-aggregation mechanism on digital mode classification.

3. ChatLLM network

In this section, we first introduce our model architecture (Section 3.1). Then we describe the feedforward process (Section 3.2), followed by the language based backpropagation mechanism (Section 3.3). Lastly, we explain the drop-out mechanism as well as the network optimization (Sections 3.4 and 3.5).

3.1. Network architecture

The ChatLLM network is a multi-layered dialogue-based language model consisting of $n-1$ fully connected layers and 1 final aggregation layer, as depicted in Fig. 2. The models at layer i are denoted as $\boldsymbol{m}_{i,1}, \boldsymbol{m}_{i,2}, \dots, \boldsymbol{m}_{i,l_i}$, where l_i represents the number of models at layer i. Adjacent layers of models communicate with each other through a leader–employee relationship, where the models at layer $i+1$ serves as the leaders for the models at layer i. A dropout and concatenation mechanism is applied after each fully connected layer. The last layer, namely the aggregation layer, is comprised of one leader model $\boldsymbol{m}_n$. It takes the aggregated input from all previous layers and generates the final output of our network.

3.2. Forward-aggregation mechanism

Unlike one standalone LLM, models in our network not only receive the question information itself, but are also given answers generated by previous layers as references. This enables the subsequent layers of models to identify the key highlights from the previous answers, resulting in more comprehensive and precise responses. Such benefits are highly applicable to many tasks such as dialogue generation. On the other hand, the instability of large language models can also be improved, as the integration of outputs from multiple members can effectively offset deviations.

We can imagine a scenario when a leader and many employees need to solve a problem. Each employee may have a unique perspective and express different ideas, but ultimately a leader will consider these ideas and make the final decision. Without considering the group's opinions, the decision would be arbitrary and imperfect. Analogously, individual dialogue-based language models, such as ChatGPT, are inherently random because of its generative structures. Therefore, with a leader evaluating the ideas generated by the employees and providing guidance, an optimal outcome can be achieved.

We define $\boldsymbol{m}_i$ as a dialogue-based language model, $\boldsymbol{m}_i^{in}$ as the input of $\boldsymbol{m}_i$, $\boldsymbol{m}^{out}$ as the output of $\boldsymbol{m}_i$, and Q represents the description of a question to be solved. Generally, we use $\boldsymbol{m}_i^{in}$ and $\boldsymbol{m}_i^{out}$ to represent the input and the output of $\boldsymbol{m}_i$. $\oplus$ means concatenation operation. For one leader $\boldsymbol{m}_{i+1}$ and its i employees $\boldsymbol{m}_1, \boldsymbol{m}_2, \dots, \boldsymbol{m}_i$, we have the following representations:

$$\begin{aligned} &\boldsymbol{m}_1^{in}, \boldsymbol{m}_2^{in}, \dots, \boldsymbol{m}_i^{in} = Q, \\ &\boldsymbol{m}_{i+1}^{in} = Q \oplus \boldsymbol{m}_1^{out} \oplus \boldsymbol{m}_2^{out} \oplus \cdots \oplus \boldsymbol{m}_i^{out}. \end{aligned} \tag{1}$$

From Eq. (1), we can see that the input of each leader is composed of the question and the output of his or her employees.

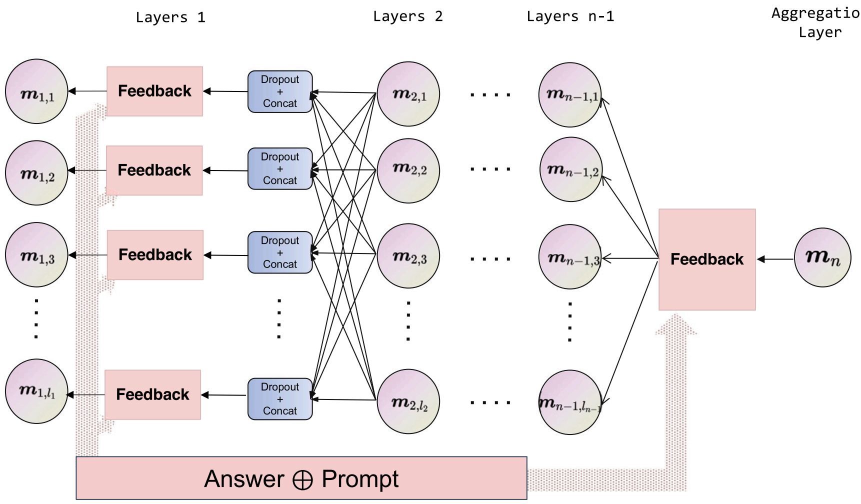


Fig. 4. Backpropagation process.

Fig. 3 shows a detailed example of the forward-aggregation mechanism on digital mode classification task.

3.3. Language based backpropagation algorithm

Traditional backpropagation algorithm calculates the gradients of the loss function with respect to the weights of a neural network, and utilizes these gradients to update the weights using an optimization algorithm such as gradient descent. We design a novel language based backpropagation algorithm to allow the ChatLLM network to learn from the incorrect samples and improve its performance over time.

Similar to real-life scenarios, a leader is able to verify the correct answer with their own earlier than their employees, in contrast to the forward-aggregation process. If a leader finds incorrection, he or she will give orders to each employee of him or her to modify their own ideas. Taking into account these feedbacks, employees will enhance their corresponding responses more effectively.

Algorithm 1 Language Based Backpropagation Mechanism

```
Input: $\{m_{i+1,j}\}$: dialogue-based large language models at layer $i+1$;
    $m_{i,*}$: an employee model at layer $i$;
    $Answer$; $Prompt$.
for $j = 1$ to $l_{i+1}$ do
    $m^{in}_{i+1,j} \leftarrow Answer \oplus Prompt$
    input $m^{in}_{i+1,j}$ to $m_{i+1,j}$;
    output $m^{out}_{i+1,j}$ from $m_{i+1,j}$;
end for
if $notmatch(m^{out}_{i,*}, Answer)$ then
    $m^{in}_{i,*} \leftarrow Answer \oplus Prompt$
    for $j = 1$ to $l_{i+1}$ do
        $m^{in}_{i,*} \leftarrow m^{in}_{i,*} \oplus m^{out}_{i+1,j}$
    end for
else
    $m^{in}_{i,*} \leftarrow Answer \oplus Prompt$
end if
input $m^{in}_{i,*}$ to $m_{i,*}$;
output $m^{out}_{i,*}$ from $m_{i,*}$;
return
```

We illustrate the language based backpropagation mechanism in Fig. 4. Specifically, after the forward process, the final output of the model will be compared with the ground-truth $Answer$. If the output is correct, the model will get the prompt ($Prompt$) "You guessed it right, remember your reasoning..." and preserve its original logic flow. If it is incorrect, the model will get the prompt ($Prompt$) "You guessed it wrong. Please speculate a possible reason why the answer is this and update your thinking". and enhance its logic optimization process. Feedback is applied throughout all layers in the network, stabilize the accurate models' states, while adjust its surmise for closer alignment with the correct output. One detailed feedback example of the language based backpropagation process is shown in Fig. 5.

If the model $\boldsymbol{m_i}$ outputs a wrong answer, it will get the following feedback as input (as shown in the upper left in Fig. 5):

$$\boldsymbol{m}^{in}_{i} = Answer \oplus Prompt \oplus \boldsymbol{m}^{out}_{i+1} \oplus \boldsymbol{m}^{out}_{i+2} \oplus \cdots \oplus \boldsymbol{m}^{out}_{n}. \tag{2}$$

Otherwise, the model will get input (shown in bottom left in Fig. 5):

$$\boldsymbol{m}^{in}_{i} = Answer \oplus Prompt. \tag{3}$$

The detailed algorithm is described in Algorithm 1. It is worth noting that the system will establish a long-term memory for each model. When the entire Language Based Backpropagation process is completed, the output of each model for the question will be saved and referred to when answering questions in the future.

3.4. Dropout mechanism

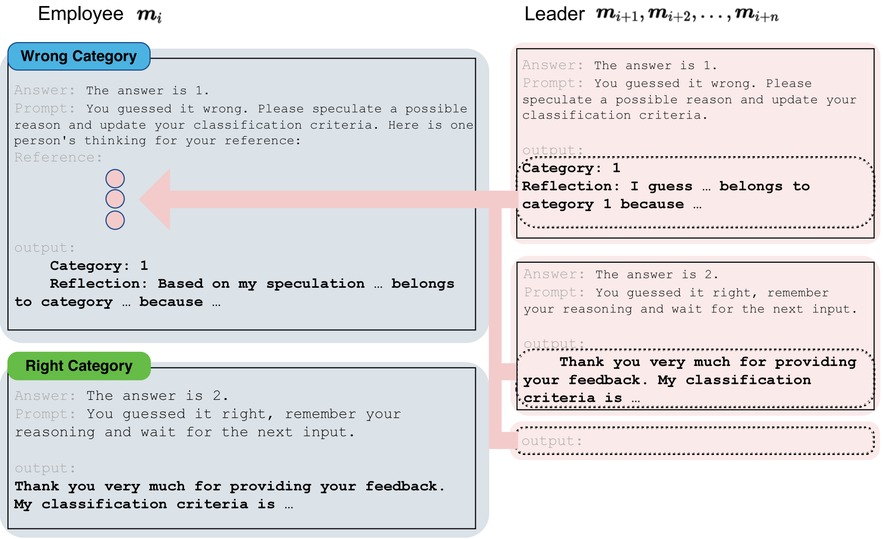
The capacity of an individual dialogue-based language model is inherently constrained. By restricting the input to an appropriate range, we can prevent these models from becoming inundated with an excessive amount of information. Additionally, the implementation of a dropout mechanism in neural networks, as described in Srivastava et al. (2014), has been shown to effectively reduce overfitting and enhance generalization performance. Therefore we devise a dropout mechanism, as described in Eq. (4).

Analogously, if a leader has too many employees, it may be difficult for them to handle all of his or her employees' ideas. Similarly, if an employee has too many leaders, it may be challenging for them to satisfy all of leaders. Therefore, based on the structure of the entire network, we allow each dialogue-based language model to randomly receive messages from only a limited number of other models, thus ensuring that the overall input is controlled within a certain range. Formally, to implement it, we calculate a random variable r whose value is 0 or 1:

$$r \sim Bernoulli(\rho), \tag{4}$$

where ρ is the rate of the number of selected models. Then the model $\boldsymbol{m}_{i+1}$ receives selected messages from the sender models:

$$\boldsymbol{m}^{in}_{\boldsymbol{i+1}} = r_1 \cdot \boldsymbol{m}^{out}_{\boldsymbol{1}} \oplus r_2 \cdot \boldsymbol{m}^{out}_{\boldsymbol{2}} \oplus \cdots \oplus r_i \cdot \boldsymbol{m}^{out}_{\boldsymbol{i}}, \tag{5}$$



```
Answer: The answer is 1.
Prompt: You guessed it wrong. Please speculate a possible
reason and update your classification criteria. Here is one
person's thinking for your reference:
Reference:

output:
    Category: 1
    Reflection: Based on my speculation … belongs
to category … because …
```

```
Answer: The answer is 2.
Prompt: You guessed it right, remember your
reasoning and wait for the next input.

output:
Thank you very much for providing your feedback.
My classification criteria is …
```

```
Answer: The answer is 1.
Prompt: You guessed it wrong. Please
speculate a possible reason and update your
classification criteria.

output:
Category: 1
Reflection: I guess … belongs to
category 1 because …
```

```
Answer: The answer is 2.
Prompt: You guessed it right, remember
your reasoning and wait for the next input.

output:
    Thank you very much for providing
your feedback. My classification
criteria is …
```

```
output:
```

Fig. 5. A feedback example of the backpropagation process in digital mode classification task.

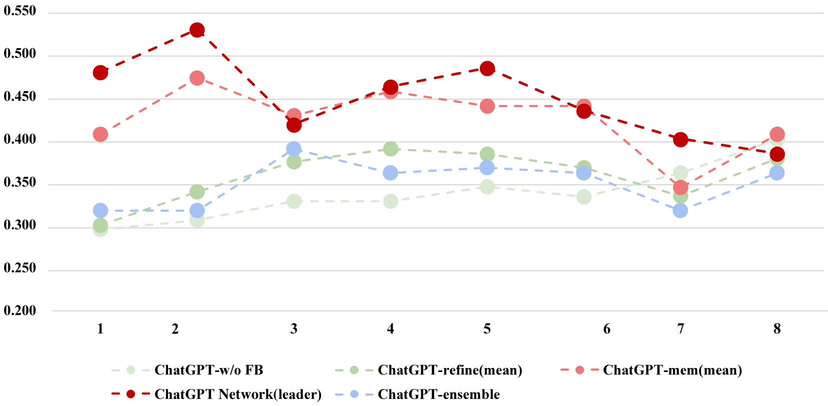


Fig. 6. Comparison of different models on accuracy along the intermediate stages.

where if $r_i = 1$, $r_i \cdot \boldsymbol{m}_i^{out}$ equals $\boldsymbol{m}_i^{out}$; otherwise, $r_i \cdot \boldsymbol{m}_i^{out}$ is a null string.

3.5. Network optimization

During the training process, individual training examples are inputted sequentially. Inspired by the Stochastic Gradient Descendent algorithm, we update the network with the language-based backpropagation mechanism for each training sample accordingly. To prevent overfitting, we employ the early stopping technique. The stopping criteria can be met by either of the two conditions: reaching a predetermined number of iterations, or the performance ceasing to show further improvement.

4. Experiments

Due to the high cost of the GPT-4 API, we choose to use ChatGPT or GPT-3.5-Turbo as the basic ChatLLM of our overall network. A collection of models is supposed to learn from and refer to each other when solving the prompted question. Noting that due to the input limitation of GPT, we design a simple network structure of ChatLLMs that does not exceed three layers, consisting of a few ChatLLMs.

We conducted three experiments to evaluate the network:First, the Digital Mode Classification Experiment was crafted to discern the model's innate learning abilities from the pre-existing implicit knowledge within a large language model, particularly its capacity to address problems in the absence of explicit rules among digits. Following that, the Sentiment Reversal Experiment was designed to underscore the network's proficiency in bolstering performance in standard NLP tasks, specifically examining its adeptness at comprehending and altering emotional contexts through sentiment reversal tasks. Lastly, the Arithmetic Reasoning Experiment served to gauge the efficacy of both two-layer and three-layer ChatLLM networks on the GSM8K (Cobbe et al., 2021) and AUQA-RAT (Ling et al., 2017) datasets, concentrating on arithmetic reasoning. Experimental details are as follows.

4.1. Digital mode classification

The experiment aims to test ChatGPT's learning ability from scratch. In the digital mode classification task, we generate a dataset consisting of different categories of digital vectors. Particularly, we categorize a three-dimensional vector (a, b, c) based on the position of the largest dimension in the vector. For example, $(1, 2, 4)$ belongs to category 3 because the largest number 4 is located in the third dimension. Since ChatGPT has no pre-existing knowledge of the task, this provides an opportunity to evaluate its inductive learning capability.

We use 24 samples for training, and conduct observations every 3 training samples. Thus we have 8 intermediate stages, wherein three vectors are prompted to the ChatLLM network between stages. The testing set comprises of 30 challenging samples that have been manually designed, which are collectively fed into the model. The outputs are

Table 1
Accuracy of different models at 8 intermediate stages.

	1	2	3	4	5	6	7	8
ChatGPT-w/o FB	0.295 (±24.10%)	0.306 (±18.77%)	0.328 (±19.76%)	0.328 (±21.74%)	0.345 (±11.73%)	0.333 (±12.68%)	0.361 (±10.81%)	0.400 (±14.90%)
ChatGPT-refine(mean)	0.300 (±23.27%)	0.339 (±26.72%)	0.374 (±16.42%)	0.389 (±11.74%)	0.383 (±16.30%)	0.367 (±29.22%)	0.334 (±31.55%)	0.378 (±18.22%)
ChatGPT-ensemble	0.317 (±12.90%)	0.317 (±12.83%)	0.389 (±6.93%)	0.361 (±7.00%)	0.367 (±15.18%)	0.361 (±21.40%)	0.317 (±19.68%)	0.361 (±10.85%)
ChatGPT-mem (mean)	0.406 (±28.63%)	0.472 (±22.32%)	**0.428** (±26.68%)	0.456 (±17.03%)	0.439 (±24.19%)	**0.439** (±25.12%)	0.344 (±20.95%)	**0.406** (±6.12%)
ChatGPT Network (leader)	**0.478** (±19.03%)	**0.528** (±13.48%)	0.417 (±16.55%)	**0.461** (±9.63%)	**0.483** (±16.74%)	0.433 (±17.54%)	**0.400** (±7.38%)	0.383 (±14.28%)

the corresponding label for each vector. No feedback is involved in the testing process.

For evaluation, we report the accuracy by checking if the categories are consistent with the pre-defined rules. We compare our ChatLLM network model with the following baselines:

ChatGPT-w/o FB: a vanilla ChatGPT takes the question and training input vectors with categories as input without further feedback.

ChatGPT-refine: a vanilla ChatGPT takes the same input as ChatGPT-w/o FB, and if the answer is incorrect, we request it to refine the answer with the instruction "*refine your answer*"

ChatGPT-ensemble: using simple voting mechanism and selecting the most frequent answer amongst three individual ChatGPTs as the consensus output.

Table 1 reports the average results of six times for all the models. From the table, we can observe that: (1) Our proposed ChatGPT network, i.e., ChatGPT Network (leader) taking the leader output as final output, significantly outperforms all the baselines. It demonstrates the enormous advantages of the ChatGPT network in terms of forward aggregation and backward feedback. (2) The mean output of the members of our ChatGPT network, i.e., ChatGPT-mem (mean), also has a significantly higher accuracy than that of the baselines, which validates that the proposed ChatLLM network model can improve individual ChatLLMs in the network. (3) The variance of the ChatGPT Network (leader) closely resembles that of the ChatGPT-ensemble, while lower than other baselines. This indicates that the output of the ChatGPT network is relatively stable, like the ChatGPT-ensemble baseline.

Fig. 6 illustrates the comparison of accuracy along the intermediate stages. We can observe that our proposed ChatLLM network generally achieves better results at all intermediate stages. We also find that the accuracy values of all the models consistently first increase to a peak, and then begin to decrease. This is due to the overfitting problem since we observe that the models gradually increase the complexity of the classification criteria after reaching the peak. Early stopping during the training process is necessary to achieve the best results.

4.2. Sentiment reversal experiment

Sentiment reversal is a typical NLP task that rewrites a given sentence by reversing its current sentiment (positive or negative) (Madaan et al., 2023).

Dataset Generation. We generate a dataset comprising 60 emotionally biased sentences, along with their corresponding sentiments, using ChatGPT.

Experimental Setting. Our objective is to examine the efficacy of the backpropagation feedback mechanism from leaders to employees in our proposed ChatLLM network. We compare ChatGPT network with an individual ChatGPT model. Note that there is no training process for all models in this experiment. We apply the backpropagation feedback mechanism during testing instead. Specifically, our experiments can be divided into two groups:

w/o FB: No backpropagation feedback is provided for any model in this setting.

Table 2
The results of sentiment reversal task.

	Win	Loss	Tie
ChatGPT w/o FB	22	26	12
ChatGPT network w/o FB	26	22	12
ChatGPT w/ FB	7	53	0
ChatGPT network w/ FB	53	7	0

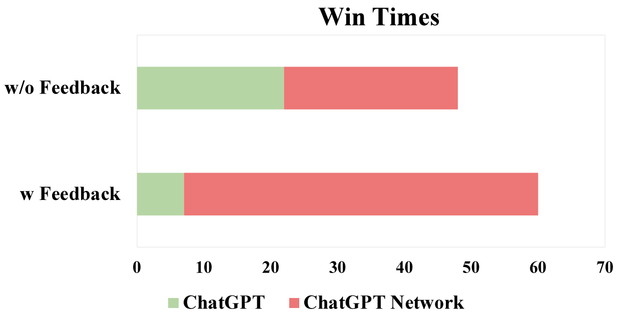


Fig. 7. Comparison of win times.

w/ FB: We direct both the baseline model ChatGPT and our ChatGPT network to augment the emotional intensity of the former output sentences, with the feedback in the way of prompt "*Make the emotionally reversed sentences more emotionally intense*". Under this setting, the baseline model ChatGPT employs self-feedback, whereas the ChatGPT network utilizes backpropagation for providing feedbacks and obtaining new results. The final output of ChatGPT network is obtained by another forward aggregation process based on new results.

Evaluation. Following Madaan et al. (2023), the evaluation process incorporates a separate ChatGPT as a judge, which is responsible for determining which group produced sentences with more intense emotions and gives the reasons. The scores assigned by the ChatGPT judge are reported in Table 2 and Fig. 7. Each superior sentence earns one point for the generating model. To ensure fairness and eliminate potential biases in the ChatGPT judge's scoring, we request the provision of a rationale for each decision as illustrated in Table 3.

The results in Table 2 and Fig. 7 reveal that without feedback, the ChatGPT network displays an marginally enhanced performance compared to an isolated ChatGPT (baseline model), attributable to its ability to summarize information. However, when feedback is applied, our ChatGPT network significantly outperforms the standalone baseline ChatGPT, showing the immense improvement of the feedback on the ChatGPT network's performance.

In Table 3, two examples are provided to illustrate the ChatGPT network's results. As we can observe that compared to ChatGPT, our proposed ChatGPT network can generate a sentence with reversal sentiment in higher emotional intensity of adjectives and in richer vocabulary.

Table 3
Sentiment reversal examples. The original inputs are as follows. Example 1: *This movie is interesting.* Example 2: *This journey is satisfying.*

	ChatGPT	ChatGPT network	Result	Reason
Example1	This movie is incredibly dull.	This movie is excruciatingly dull.	ChatGPT network wins	Both sentences express negative emotions towards the movie, but “excruciatingly dull” implies a stronger degree of negative feeling compared to “incredibly dull.”
Example2	This journey is completely unfulfilling.	This journey is soul-crushingly and utterly unfulfilling	ChatGPT network wins	The addition of “soul-crushingly and utterly” intensifies the negative emotion of the sentence,making it feel more impactful and powerful.

Table 4
Comparison of different models based on GSM8K and AUQA-RAT scores, with improvement percentages compared to Zero-shot CoT.

Model	GSM8K	AUQA-RAT
Zero-shot CoT	0.745	0.579
2-layers Network (w/o FB)	0.804 (+7.92%)	0.594 (+2.59%)
3-layers Network (w/o FB)	0.814 (+9.26%)	0.610 (+5.35%)
2-layers Network	0.842 (+13.02%)	0.628 (+8.46%)
3-layers Network	**0.867** (+16.38%)	**0.639** (+10.36%)

4.3. Arithmetic reasoning experiment

In this experiment, we evaluated the performance of two-layer and three-layer ChatLLM networks on the GSM8K (Cobbe et al., 2021) and AUQA-RAT (Ling et al., 2017) datasets. GSM8K consists of 8.5K high-quality mathematical problems, all of which are created by human writers. These problems require 2 to 8 steps to solve, with the main solution method being a series of basic calculations using fundamental arithmetic operations to arrive at the final answer. The AQUA-RAT dataset consists of about 100,000 algebraic word problems with natural language rationales. Both of these datasets are common benchmarks for arithmetic reasoning. The goal was to compare the performance of different network architectures on arithmetic reasoning tasks. To comprehensively assess the model performance, we used 2 arithmetic reasoning benchmarks for testing.

We compared the performance of GPT-3.5-Turbo in a zero-shot chain-of-thought (CoT) setting and evaluated the two-layer and three-layer ChatLLM networks with and without feedback mechanism (FB).

The ChatLLM Network with two layers is composed of a stack of GPT-3.5-Turbo models, where the first layer consists of three GPT-3.5-Turbo instances, and the second layer comprises a single GPT-3.5-Turbo model. For the three-layer ChatLLM Network, the configuration is tiered with the first layer having three GPT-3.5-Turbo models, the second layer having two, and the topmost layer with just one GPT-3.5-Turbo model. This hierarchical structure allows for an increase in complexity and depth of processing as information moves through the network. The experimental results are shown in Table 4.

Analysis and discussion. The experimental results indicate that the ChatLLM network, even without the feedback mechanism, significantly outperforms the baseline GPT-3.5-Turbo model on both datasets. Notably, the three-layer ChatLLM network achieved scores of 0.814 and 0.610 on the GSM8K and AUQA-RAT datasets, respectively.

With the feedback mechanism enabled, the performance of the ChatLLM networks improved further. The three-layer ChatLLM network achieved the highest performance, with scores of 0.867 and 0.639 on the GSM8K and AUQA-RAT datasets, respectively. This demonstrates that both increasing the number of network layers and incorporating a feedback mechanism significantly enhance the model’s arithmetic reasoning capabilities.

These experimental results validate the effectiveness of the ChatLLM network in handling arithmetic reasoning tasks. They also highlight the potential for further performance improvements through structural enhancements and the introduction of feedback mechanisms.

5. Limitations

Due to the limited capabilities of current dialogue-based language models, especially in processing large-scale numerical inputs, our network may not demonstrate absolute superiority in certain scenarios. Furthermore, the absence of an efficacious communication mechanism among different dialogue language models precludes larger scales of collaboration, restricting our present research to few members of the network.

6. Conclusion

In this work, we propose a novel ChatLLM network that allows multiple dialogue-based language models to interact, provide feedback, and think together. Specifically, individual instances of ChatLLM in the network may possess distinct perspectives towards the same problem, and by consolidating these diverse viewpoints via a separate ChatLLM, the ChatLLM network system can conduct decision-making more objectively and comprehensively. The optimization of the network is carried out based on a novel language-based backpropagation mechanism. We evaluate the network’s performance through experiments on three tasks, demonstrating the effectiveness and superiority of the ChatLLM network.

While acknowledging the existence of limitations, such as the lack of a unified mechanism for communication between the models, we believe that our research will serve as a foundational work to provide valuable insights to guide future endeavor in the field. As part of our ongoing efforts, we plan to develop and implement a global strategy for assigning distinct identities to each model in the network, thereby ensuring that each model performs its exclusive task and enhancing the traceability of inter-model communication.

CRediT authorship contribution statement

Rui Hao: Writing – review & editing, Writing – original draft, Software, Methodology. **Linmei Hu:** Writing – original draft, Supervision, Methodology. **Weijian Qi:** Writing – original draft, Formal analysis. **Qingliu Wu:** Formal analysis. **Yirui Zhang:** Writing – original draft. **Liqiang Nie:** Supervision.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgments

This work was supported by the National Science Foundation of China (No. 62276029), CCF-Zhipu. AI Large Model Fund (No. 202217), Beijing Institute of Technology Research Fund Program for Young Scholars, China (No. 6120220261), and CIPSC-SMP-Zhipu Large Model Cross-Disciplinary Fund.

References

Bradley Knox, W., Stone, P., 2008. TAMER: Training an agent manually via evaluative reinforcement. In: 2008 7th IEEE International Conference on Development and Learning. pp. 292–297.

Brown, T., Mann, B., Ryder, N., Subbiah, M., Kaplan, J.D., Dhariwal, P., Neelakantan, A., Shyam, P., Sastry, G., Askell, A., Agarwal, S., Herbert-Voss, A., Krueger, G., Henighan, T., Child, R., Ramesh, A., Ziegler, D., Wu, J., Winter, C., Hesse, C., Chen, M., Sigler, E., Litwin, M., Gray, S., Chess, B., Clark, J., Berner, C., McCandlish, S., Radford, A., Sutskever, I., Amodei, D., 2020. Language models are few-shot learners. In: Advances in Neural Information Processing Systems, vol. 33, Curran Associates, Inc., pp. 1877–1901.

Chan, C.M., Chen, W., Su, Y., Yu, J., Xue, W., Zhang, S., Fu, J., Liu, Z., 2024. ChatEval: Towards better LLM-based evaluators through multi-agent debate. In: The Twelfth International Conference on Learning Representations.

Chang, E.Y., 2024. SocraSynth: Multi-LLM reasoning with conditional statistics. arXiv Preprint, arXiv:2402.06634.

Chen, P.L., Chang, C.-S., 2023. InterAct: Exploring the potentials of ChatGPT as a cooperative agent. arXiv Preprint, arXiv:2308.01552.

Chen, W., Su, Y., Zuo, J., Yang, C., Yuan, C., Chan, C.M., Yu, H., Lu, Y., Hung, Y.H., Qian, C., Qin, Y., Cong, X., Xie, R., Liu, Z., Sun, M., Zhou, J., 2024. AgentVerse: Facilitating multi-agent collaboration and exploring emergent behaviors. In: The Twelfth International Conference on Learning Representations.

Chen, Z., Zhou, K., Zhang, B., Gong, Z., Zhao, X., Wen, J.R., 2023. ChatCoT: Tool-augmented chain-of-thought reasoning on chat-based large language models. In: Findings of the Association for Computational Linguistics. EMNLP 2023, Association for Computational Linguistics, Singapore, pp. 14777–14790.

Cobbe, K., Kosaraju, V., Bavarian, M., Chen, M., Jun, H., Kaiser, L., Plappert, M., Tworek, J., Hilton, J., Nakano, R., Hesse, C., Schulman, J., 2021. Training verifiers to solve math word problems. arXiv Preprint, arXiv:2110.14168.

Devlin, J., Chang, M.W., Lee, K., Toutanova, K., 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In: Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers). Association for Computational Linguistics, pp. 4171–4186.

Du, Y., Li, S., Torralba, A., Tenenbaum, J.B., Mordatch, I., 2024. Improving factuality and reasoning in language models through multiagent debate. In: Proceedings of the 41st International Conference on Machine Learning. In: Proceedings of Machine Learning Research, vol. 235, PMLR, pp. 11733–11763.

Guo, T., Chen, X., Wang, Y., Chang, R., Pei, S., Chawla, N.V., Wiest, O., Zhang, X., 2024. Large language model based multi-agents: A survey of progress and challenges. In: Proceedings of the Thirty-Third International Joint Conference on Artificial Intelligence. IJCAI-24, International Joint Conferences on Artificial Intelligence Organization, pp. 8048–8057.

Hong, S., Zhuge, M., Chen, J., Zheng, X., Cheng, Y., Wang, J., Zhang, C., Wang, Z., Yau, S.K.S., Lin, Z., Zhou, L., Ran, C., Xiao, L., Wu, C., Schmidhuber, J., 2024. MetaGPT: Meta programming for a multi-agent collaborative framework. In: The Twelfth International Conference on Learning Representations.

Irving, G., Christiano, P., Amodei, D., 2018. AI safety via debate. arXiv Preprint, arXiv:1805.00899.

Li, G., Hammoud, H., Itani, H., Khizbullin, D., Ghanem, B., 2023. CAMEL: Communicative agents for "mind" exploration of large language model society. In: Advances in Neural Information Processing Systems, vol. 36, Curran Associates, Inc., pp. 51991–52008.

Ling, W., Yogatama, D., Dyer, C., Blunsom, P., 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 (Volume 1: Long Papers). Association for Computational Linguistics, pp. 158–167.

Madaan, A., Tandon, N., Gupta, P., Hallinan, S., Gao, L., Wiegreffe, S., Alon, U., Dziri, N., Prabhumoye, S., Yang, Y., Gupta, S., Majumder, B.P., Hermann, K., Welleck, S., Yazdanbakhsh, A., Clark, P., 2023. Self-refine: Iterative refinement with self-feedback. In: Advances in Neural Information Processing Systems, vol. 36, Curran Associates, Inc., pp. 46534–46594.

Miao, N., Teh, Y.W., Rainforth, T., 2024. SelfCheck: Using LLMs to zero-shot check their own step-by-step reasoning. In: The Twelfth International Conference on Learning Representations.

Ouyang, L., Wu, J., Jiang, X., Almeida, D., Wainwright, C., Mishkin, P., Zhang, C., Agarwal, S., Slama, K., Ray, A., Schulman, J., Hilton, J., Kelton, F., Miller, L., Simens, M., Askell, A., Welinder, P., Christiano, P.F., Leike, J., Lowe, R., 2022. Training language models to follow instructions with human feedback. In: Advances in Neural Information Processing Systems, vol. 35, Curran Associates, Inc., pp. 27730–27744.

Press, O., Zhang, M., Min, S., Schmidt, L., Smith, N., Lewis, M., 2023. Measuring and narrowing the compositionality gap in language models. In: Findings of the Association for Computational Linguistics. EMNLP 2023, Association for Computational Linguistics, Singapore, pp. 5687–5711.

Radford, A., Narasimhan, K., 2018. Improving language understanding by generative pre-training. URL: https://cdn.openai.com/research-covers/language-unsupervised/language_understanding_paper.pdf. (Accessed 06 December 2024).

Radford, A., Wu, J., Child, R., et al., 2019. Language models are unsupervised multitask learners. OpenAI Blog 1 (8), URL: https://cdn.openai.com/better-language-models/language_models_are_unsupervised_multitask_learners.pdf. (Accessed 06 December 2024).

Raffel, C., Shazeer, N., Roberts, A., Lee, K., Narang, S., Matena, M., Zhou, Y., Li, W., Liu, P.J., 2020. Exploring the limits of transfer learning with a unified text-to-text transformer. J. Mach. Learn. Res. 21 (140), 1–67.

Shinn, N., Cassano, F., Gopinath, A., Narasimhan, K., Yao, S., 2023. Reflexion: language agents with verbal reinforcement learning. In: Advances in Neural Information Processing Systems, vol. 36, Curran Associates, Inc., pp. 8634–8652.

Srivastava, N., Hinton, G., Krizhevsky, A., Sutskever, I., et al., 2014. Dropout: A simple way to prevent neural networks from overfitting. J. Mach. Learn. Res. 15 (56), 1929–1958.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A.N., Kaiser, L.u., Polosukhin, I., 2017. Attention is all you need. In: Advances in Neural Information Processing Systems, vol. 30, Curran Associates, Inc..

Wang, L., Ma, C., Feng, X., Zhang, Z., Yang, H., Zhang, J., Chen, Z., Tang, J., Chen, X., Lin, Y., Zhao, W.X., Wei, Z., Wen, J., 2024. A survey on large language model based autonomous agents. Front. Comput. Sci. (ISSN: 2095-2236) 18 (6).

Wei, J., Bosma, M., Zhao, V., Guu, K., Yu, A.W., Lester, B., Du, N., Dai, A.M., Le, Q.V., 2022a. Finetuned language models are zero-shot learners. In: International Conference on Learning Representations.

Wei, J., Wang, X., Schuurmans, D., Bosma, M., ichter, b., Xia, F., Chi, E., Le, Q.V., Zhou, D., 2022b. Chain-of-thought prompting elicits reasoning in large language models. In: Advances in Neural Information Processing Systems, vol. 35, Curran Associates, Inc., pp. 24824–24837.

Xi, Z., Chen, W., Guo, X., He, W., Ding, Y., Hong, B., Zhang, M., Wang, J., Jin, S., Zhou, E., Zheng, R., Fan, X., Wang, X., Xiong, L., Zhou, Y., Wang, W., Jiang, C., Zou, Y., Liu, X., Yin, Z., Dou, S., Weng, R., Cheng, W., Zhang, Q., Qin, W., Zheng, Y., Qiu, X., Huang, X., Gui, T., 2023. The rise and potential of large language model based agents: A survey. arXiv Preprint, arXiv:2309.07864.