Long-form Hallucination Detection with Self-elicitation

Zihang Liu[1], **Jiawei Guo**[1], **Hao Zhang**[2], **Hongyang Chen**[3],
Jiajun Bu[1], **Haishuai Wang**[1,*]
[1]Zhejiang Key Laboratory of Accessible Perception and Intelligent Systems,
College of Computer Science and Technology, Zhejiang University
[2]Alibaba Health [3]Zhejiang Lab
[1]{lzhmark,guojiawei,bjj,haishuai.wang}@zju.edu.cn,
[2]mark.zhangh@alibaba-inc.com, [3]hongyang@zhejianglab.com

Abstract

While Large Language Models (LLMs) have exhibited impressive performance in generating long-form content, they frequently present a hazard of producing factual inaccuracies or hallucinations. An effective strategy to mitigate this hazard is to leverage off-the-shelf LLMs to detect hallucinations after the generation. The primary challenge resides in the comprehensive elicitation of the intrinsic knowledge acquired during their pre-training phase. However, existing methods that employ multi-step reasoning chains predominantly fall short of addressing this issue. Moreover, since existing methods for hallucination detection tend to decompose text into isolated statements, they are unable to understand the contextual semantic relations in long-form content. In this paper, we study a novel concept, self-elicitation, to leverage self-generated thoughts derived from prior statements as catalysts to elicit the expression of intrinsic knowledge and understand contextual semantics. We present a framework, SelfElicit, to integrate self-elicitation with graph structures to effectively organize the elicited knowledge and facilitate factual evaluations. Extensive experiments on five datasets in various domains demonstrate the effectiveness of self-elicitation and the superiority of our proposed method.

1 Introduction

Large Language Models (LLMs), pre-trained on massive text corpora and fine-tuned to follow human instructions (Bai et al., 2023; Touvron et al., 2023; GLM et al., 2024; AI@Meta, 2024), have shown remarkable performance to generate long-form content that consists of multiple coherent sentences (Wei et al., 2022). However, there remains a concern regarding their tendency to generate factual (external) hallucinations (Bang et al., 2023), producing sentences that appear plausible but are factually unsupported (Huang et al., 2023). This issue undermines their reliability in real-world scenarios where factually accurate responses are expected (Wei et al., 2024). For example, a model-generated non-factual statement, *"Gliclazide can be taken at any time of the day, regardless of whether it is on an empty stomach or after meals"*, could mislead patients into taking medication at inappropriate times, as the medication is actually recommended to be taken with food (NHS, 2024). An important strategy to alleviate hallucinations is to detect hallucinations after generation (Lee et al., 2023b; Manakul et al., 2023; Mishra et al., 2024; Guan et al., 2024).

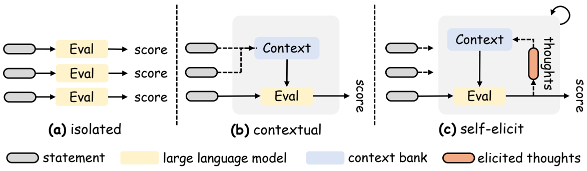


Figure 1: Schematic illustration of hallucination detection in long-form content. **(a)** Statements are isolatedly evaluated. **(b)** Prior statements are incorporated as context. We investigate **(c)** how prior self-generated thoughts can elicit models' intrinsic knowledge.

The detection of hallucinations in large language models has been approached through various methods, such as retrieval-based techniques (Min et al., 2023; Li et al., 2023b; Xia et al., 2024; Wei et al., 2024; Yue et al., 2024; Sansford et al., 2024) and probe-based approaches (Li et al., 2023a; Zhang et al., 2024a; Chuang et al., 2024; Wang et al., 2024a). However, these methods rely heavily on external databases or training corpora, which may not always be available. Consequently, many studies have shifted toward leveraging the intrinsic capabilities of pre-trained, off-the-shelf LLMs. The central challenge lies in effectively eliciting models' internal knowledge for hallucination detection. For example, Zhao et al. (2024) and Wang et al. (2024b) prompt the models to verbally express knowledge through chain-of-thought reasoning. Manakul et al.

[*]Corresponding author: Haishuai Wang

Findings of the Association for Computational Linguistics: ACL 2025, pages 4082–4100
July 27 - August 1, 2025 ©2025 Association for Computational Linguistics

(2023); Mündler et al. (2024); Miao et al. (2024) prompt the model to generate statements from various perspectives and quantify their semantic consistencies. Other works (Kang et al., 2023; Dhuliawala et al., 2024; Farquhar et al., 2024; Setty and Setty, 2024) ask the model to answer verification questions corresponding to the factoids. While theoretically insightful, these multi-step elicitation procedures often encounter limitations in practice. They either tend to express knowledge with low relevance or are prone to accumulated inaccuracies when extracting triples or constructing verification questions, therefore requiring labor-intensive, meticulous design of prompts or samples to ensure accuracy and performance.

Additionally, the long-form content generated by LLMs consists of multiple semantically related sentences that exhibit logical relationships (Quan et al., 2024; Que et al., 2024), such as coherence, comparison, and causality. For instance, the preceding statement, *"Gliclazide is an oral hypoglycemic medication"* and the subsequent statement, *"It is suitable for adult type 2 diabetes patients whose blood sugar cannot be adequately controlled by diet alone"*, demonstrate logical coherence and progression. The first statement introduces the category and function of the medication, while the second statement further elaborates on its specific medical application. However, existing long-form hallucination detection methods (Zhang et al., 2020; Min et al., 2023; Wei et al., 2024; Li et al., 2024b) generally decompose the long-form text into isolated statements for individual fact-checking (Figure 1(a)), overlooking the contextual semantic relationships and limiting their long-form understanding. Incorporating contextual information into models (Figure 1(b)) can provide a more natural, coherent chain of meanings, enhancing the understanding and evaluation of successive statements.

In this work, we present **SelfElicit**, an integrated framework designed to effectively elicit a model's intrinsic knowledge and utilize semantic relations for hallucination detection in long-form content. Specifically, it evaluates the factuality of each statement and elicits relevant knowledge through reflection conditioned on the evaluation. These elicited thoughts are then incorporated into subsequent evaluations as contextual information (Figure 1(c)), forming an iterative process where the evaluation and elicitation interact, namely self-elicitation. To mitigate hallucinations arising during the self-elicit process, we integrate a knowledge hypergraph with self-elicitation to facilitate knowledge retention, deduplication, and resolution of inconsistencies. Extensive experiments on five datasets in various domains demonstrate that self-elicitation can act as an effective catalyst to improve both the factuality and diversity of models' knowledge expression and our framework outperforms existing methods for long-form hallucination detection. To sum up, our contributions include:

- We study a novel concept of self-eliciting large language models for hallucination detection. We show that using self-generated thoughts from prior statements as catalysts effectively facilitates intrinsic knowledge expression and hallucination detection.
- We propose a new framework, SelfElicit, for long-form hallucination detection, which synergizes the self-elicitation mechanism with contextual semantic relation understanding. A knowledge hypergraph is integrated to organize the elicited knowledge and alleviate hallucination snowballing.
- Our framework consistently demonstrates superior performance in long-form hallucination detection on datasets with various domains with different language models. We further show that self-elicitation enhances knowledge expression with better factuality and diversity.

2 Preliminaries

2.1 Task

In this paper, we investigate the task of long-form hallucination detection, which aims to detect underlying factual incorrectness from a given long-form content. The hallucination detection uses the intrinsic capabilities of LLMs without external databases, fine-tuning, or neural probes.

Given a user query Q and a candidate response R that includes several sentences $\{r_1, r_2, \cdots\}$, the long-form hallucination detection task is to identify whether there is any factual incorrectness in each sentence and the entire response. Formally,

$$f_{LM}^{S} : (Q, r_1, r_2, \cdots) \rightarrow \hat{y}_1, \hat{y}_2, \cdots$$
$$f_{LM}^{R} : (Q, R) \rightarrow \hat{Y}$$

where f_{LM}^{S} and f_{LM}^{R} respectively refer to sentence-wise and response-wise algorithms with language models. $\hat{y}_i$ and $\hat{Y}$ are binary predictions for each sentence r_i and for the response R, respectively, where positive values refer to hallucinated and negative values refer to factual.

2.2 Knowledge Hypergraph

A knowledge hypergraph is a memory bank that stores and describes the relationships of knowledge with a graph structure. Each vertex v refers to an entity. Each hyperedge e connecting any number of vertices refers to a piece of knowledge relating to these entities (denoted as e.vertices). For example, an edge *"The mechanism of Gliclazide is to lower blood glucose by stimulating pancreatic β-cells to secrete insulin"* connects vertices *"Gliclazide"*, *"blood glucose"*, *"pancreatic β-cells"*, and *"insulin"* since the statement is directly related to these concepts. We denote the graph as $\mathcal{G} = (\mathbb{V}, \mathbb{E})$, where $\mathbb{V}$ and $\mathbb{E}$ respectively refer to the vertex set and the edge set. Compared with conventional knowledge graphs constructed by triples symbolizing knowledge regarding only two entities, a hyperedge interconnects any number of entities and thus is favorable for describing complex knowledge (Chen et al., 2024).

3 Methodology

Figure 2 provides an overview of our framework. Given long-form content to be fact-checked, we first extract important entities and statements corresponding to factoids to be checked. We then present the framework along with a knowledge hypergraph to iteratively evaluate the factuality of each statement via ①retrieving contextual information from the graph, ②evaluating the factuality based on calibration, ③reflecting for intrinsic knowledge elicitation, ④resolving inconsistencies suggesting induced hallucinations and ⑤updating the graph to retain the elicited thoughts.

3.1 Statement Extraction

Extending previous works (Min et al., 2023; Dammu et al., 2023; Wei et al., 2024) extracting statements via LLMs, we first propose to identify named entities before the extractions to alleviate information missing. Formally,

$$V_1, V_2, \cdots = \mathrm{LM}(\mathcal{P}_{ett}, r_1, r_2, \cdots), \quad (1)$$

$$\hat{s}_{i,1}, \hat{s}_{i,2}, \cdots = \mathrm{LM}(\mathcal{P}_{ext}, V_i, r_i), \quad (2)$$

where V_i is the entity set corresponding to sentence r_i. $\hat{s}_{i,j}$ refers to the j-th statements extracted from sentence r_i concerning entities V_i. $\mathcal{P}_{ett}$ and $\mathcal{P}_{ext}$ are the prompts for entity and statement extraction with domain expertise (shown in Appendix F). We concatenate all $\hat{s}_{i,j}$ to obtain the candidate statement list $\{s_1, s_2, \cdots\}$.

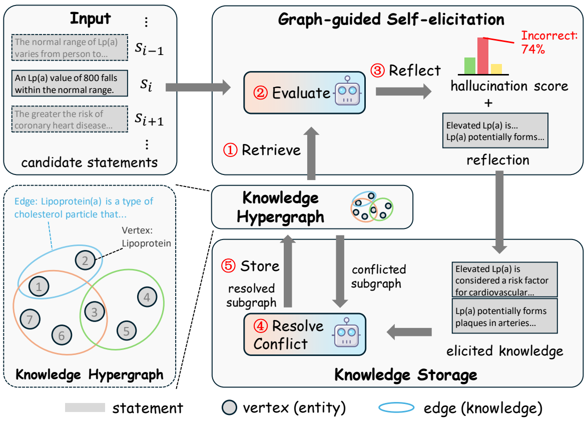

Input
The normal range of Lp(a) varies from person to...
s_{i-1}
An Lp(a) value of 800 falls within the normal range.
s_i
The greater the risk of coronary heart disease...
s_{i+1}
candidate statements
Graph-guided Self-elicitation
② Evaluate
③ Reflect
Incorrect: 74%
hallucination score
+
Elevated Lp(a) is...
Lp(a) potentially forms...
reflection
① Retrieve
Knowledge Hypergraph
Edge: Lipoprotein(a) is a type of cholesterol particle that...
Vertex: Lipoprotein
1
2
3
4
5
6
7
Knowledge Hypergraph
⑤ Store
resolved subgraph
conflicted subgraph
④ Resolve Conflict
Elevated Lp(a) is considered a risk factor for cardiovascular...
Lp(a) potentially forms plaques in arteries...
elicited knowledge
Knowledge Storage
statement
vertex (entity)
edge (knowledge)

Figure 2: The overall framework. For each candidate statement extracted from long-form content, we iteratively use a Graph-guided Self-elicitation process for hallucination detection and knowledge elicitation and a Knowledge Storage process to adaptively memorize the elicited knowledge for subsequent evaluations.

We then construct the initial knowledge hypergraph as $\mathcal{G}_0 = (\mathbb{V}, \mathbb{E}_0)$, whose vertex set includes all identified entities, i.e., $\mathbb{V} = V_1 \cup V_2 \cup \cdots$, and edge set is empty, i.e., $\mathbb{E}_0 = \varnothing$.

3.2 Graph-guided Self-elicitation

Knowledge Sampling. Given graph $\mathcal{G}_{i-1} = (\mathbb{V}, \mathbb{E}_{i-1})$ retaining self-generated thoughts from prior evaluations of $\{s_1, s_2, \cdots, s_{i-1}\}$, a graph sampling procedure is conducted to retrieve contextual information and intermediate thoughts related to current statement s_i. Specifically, entities relevant to s_i are identified by string matching, i.e., $\mathbb{V}_i = \{v_j | v_j \text{ in } s_i, v_j \in \mathbb{V}\}$. Then, sub-graphs with various relevance degrees are extracted using the combinations of the entities as queries:

$$\hat{\mathbb{V}}_i(k) = \mathrm{Combine}(\mathbb{V}_i, k), \quad (3)$$

$$\hat{\mathbb{E}}_i(k) = \{e | e.\text{vertices} == \hat{\mathbb{V}}_i(k), e \in \mathbb{E}_{i-1}\}, \quad (4)$$

$$\hat{\mathbb{E}}_i = \cup\{\hat{\mathbb{E}}_i(k) | \alpha \leq k \leq \beta\}, \quad (5)$$

where Combine$(\cdot)$ refers to k-combinations of elements $\mathbb{V}_i$. α and β are hyperparameters balancing the relevance and diversity ranges. Lower k suggests a wider scope with richer diversity, while higher k refers to a more precise matching strategy for stronger relevance (more studies in Appendix D.2). Finally, all statements corresponding to the sampled edges $\hat{\mathbb{E}}_i$ are appended to obtain context C_i.

Fact-Checking via Calibration. Calibration-based fact-checking has shown stable and competitive performance in previous literature (Kadavath et al., 2022; Manakul et al., 2023; Zhao et al., 2024; Tian et al., 2024). Similarly, we prompt the models to evaluate the factuality of a given statement s_i by asking whether the statement is `True`, `False`, or `Not Sure` and obtain the normalized logit of `False` at the first output token as the hallucination score $\hat{p}_i$. The sampled statements C_i are prefixed before s_i to provide contextual information for a better understanding of semantic relationships.

Calibration-based Elicitation. Efforts have been made to elicit the intrinsic knowledge (Petroni et al., 2019) such as cloze (Miao et al., 2024; Mündler et al., 2024), validation questions (Manakul et al., 2023; Dhuliawala et al., 2024), or chain-of-thought (Weller et al., 2024; Zhao et al., 2024). However, we observe that these methods are prone to inconsistent outputs or accumulated inaccuracies during multi-step open-ended generations and neglect the semantic relations in context.

To this end, we present an approach that guides the models to verbally express relevant intrinsic knowledge by elaborating on their calibrated first output token. We found in our experiments that this approach provides more consistent reasoning and better contextual understanding. Formally,

$$o_i^{eval}, o_i^{refl} = \text{LM}(C_i, \mathcal{P}_{eval}, s_i), \qquad (6)$$

where $\mathcal{P}_{eval}$ is the evaluation and reflection prompt (full prompt in Figure 11, Appendix F).

```
Context: {context C_i}
Description: {candidate s_i}
Is the above description:
A True    B False    C Not Sure
Choose your option and explain why:
```

o_i^{eval} refers to the first output token, where we obtain the normalized hallucination score $\hat{p}_i$. o_i^{refl} refers to the reflection sentences with objective knowledge extracted via rules or prompting from the models' outputs, which consist of detailed elaborations, subsequent deductions, factoids, and subjective opinions.

3.3 Elicited Knowledge Storage

Graph Updating. After eliciting intrinsic knowledge conditioned on the statement, we store it in the graph to provide contextual information for subsequent evaluations and handle potential knowledge inconsistencies. Specifically, for each sentence extracted from reflection, i.e., $c_{i,j} \in o_i^{refl}$, a corresponding new edge $\tilde{e}_{i,j}$ is created with verbally matched entities as its vertices, i.e., $\tilde{\mathbb{V}}_{i,j} = \{v \mid v \text{ in } c_{i,j},\ v \in \mathbb{V}\}$. All new edges obtained from reflection are dentoed as:

$$\tilde{\mathbb{E}}_i = \{\tilde{e}_{i,j} \mid \tilde{e}_{i,j}.\text{vertices} == \tilde{\mathbb{V}}_{i,j}\}, \qquad (7)$$

where j is the sentence index within o_i^{refl}. We then iteratively merge each new edge in $\tilde{\mathbb{E}}_i$ into graph $\mathcal{G}_{i-1}$ to obtain the updated graph $\mathcal{G}_i$:

$$\mathbb{E}_i = \text{Merge}(\mathbb{E}_{i-1}, \tilde{\mathbb{E}}_i). \qquad (8)$$

Consistency Checking. In some cases, models might produce hallucinations during the reflection processes, especially when reasoning on ambiguous or unfamiliar statements. Previous works (Mündler et al., 2024; Yehuda et al., 2024; Farquhar et al., 2024) have shown that models' fabricated statements are less likely to be self-consistent. To alleviate the hallucination snowballing (Zhang et al., 2023a), we conduct contrasts across reflections within each elicitation and across elicitations based on self-consistency (Kuhn et al., 2023; Farquhar et al., 2024) with the knowledge hypergraph.

In each contrast procedure, we first identify the conflicted statement, $\tilde{e} \in \tilde{\mathbb{E}}_i$, that shares identical vertices with existing edges, i.e., $\tilde{e}.\text{vertices} == e.\text{vertices}, \exists e \in \mathbb{E}_{i-1}$. Next, a Natural Language Inference (NLI) process categorizes the semantic relationships into three types: Entail (statements are identical in meaning, leading to the replacement of the original statement), Contradict (statements have directly opposite meanings), or Neutral (statements describe different entities or aspects and are both retained) with the following prompt (full prompt in Figure 12, Appendix F).

```
Determine the relationship between two sentences.
[entail]: Identical content, describing the same
aspect of the same object.
[contradict]: Directly opposite content about the
same aspect of the same object.
[neutral]: Different objects or aspects, allowing
coexistence.
Sentence A: {sentence_A}
Sentence B: {sentence_B}
```

If contradicted, a revision and resolution of the conflict are conducted with the following prompt (full prompt in Figure 13, Appendix F).

```
Sentence A: {sentence_A}
Sentence B: {sentence_B}
Two sentences describe the same subject but are
contradictory. Determine which is more accurate
based on logic and factuality.
```

Table 1: Full hallucination detection results. S: sentence-wise metrics. R: response-wise metrics. **Red**: the best. Blue: the second best. Higher metrics are better.

LLM	Metric	**SelfElicit**		IO		ContextIO		HistoryIO		CoT		CoVE		FaR		SelfChkGPT		ChatProtect	
		S	R	S	R	S	R	S	R	S	R	S	R	S	R	S	R	S	R
WikiBio (biography)																			
Qwen	AUC	**0.594**	0.653	0.527	0.628	0.587	0.522	0.543	0.614	0.500	0.566	0.527	0.524	0.543	0.508	0.539	0.639	0.512	**0.657**
Llama2		**0.578**	0.707	0.516	0.559	0.534	0.534	0.477	0.540	0.534	0.531	0.553	0.636	0.506	0.522	0.572	**0.708**	0.517	0.704
FActScore (biography)																			
Qwen	AUC	**0.524**	0.557	0.497	0.455	0.504	0.359	0.486	0.344	0.510	**0.572**	0.486	0.485	0.500	0.462	0.495	0.481	0.495	0.382
Llama2		0.534	**0.545**	0.494	0.345	0.475	0.404	0.526	0.353	0.498	0.471	0.481	0.427	0.453	0.380	**0.536**	0.471	0.496	0.461
HaluEval2 (education, finance, science)																			
Qwen	AUC	**0.789**	**0.564**	0.729	0.516	0.680	0.460	0.780	0.532	0.599	0.516	0.672	0.513	0.757	0.557	0.631	0.524	0.528	0.494
Llama2		0.581	**0.545**	0.513	0.529	0.502	0.493	0.516	0.491	0.598	0.499	0.593	0.505	0.596	0.489	**0.601**	0.498	0.560	0.541
MedHallu-zh (medicine)																			
Qwen	$F1$	**0.269**	**0.475**	0.187	0.441	0.191	0.430	0.238	0.453	0.192	0.402	0.165	0.395	0.207	0.441	0.085	0.395	0.085	0.395
	AUC	**0.810**	**0.671**	0.771	0.598	0.760	0.603	0.782	0.653	0.638	0.571	0.597	0.548	0.763	0.613	0.500	0.500	0.512	0.517
GLM	$F1$	**0.228**	**0.445**	0.182	0.421	0.153	0.424	0.213	0.435	0.131	0.395	0.170	0.423	0.139	0.405	0.085	0.395	0.134	0.395
	AUC	**0.798**	**0.622**	0.756	0.598	0.733	0.582	0.781	0.614	0.564	0.527	0.661	0.567	0.702	0.554	0.494	0.500	0.611	0.558
MedHallu-en (medicine)																			
Qwen	$F1$	**0.242**	0.463	0.182	0.436	0.168	0.443	0.233	**0.472**	0.192	0.395	0.085	0.395	0.187	0.445	0.226	0.428	0.085	0.395
	AUC	**0.803**	0.656	0.762	0.622	0.743	0.614	0.779	**0.659**	0.596	0.570	0.500	0.498	0.763	0.630	0.682	0.623	0.505	0.505
Qwen2	$F1$	**0.282**	**0.479**	0.275	0.466	0.247	0.460	0.254	0.456	0.211	0.422	0.259	0.440	0.217	0.447	0.232	0.444	0.087	0.395
	AUC	**0.820**	**0.667**	0.805	0.665	0.802	0.661	0.811	0.656	0.636	0.595	0.672	0.614	0.784	0.640	0.675	0.636	0.523	0.537
Llama2	$F1$	**0.181**	0.408	0.137	0.410	0.139	0.407	0.133	**0.413**	0.142	0.395	0.085	0.395	0.140	0.411	0.103	0.397	0.136	0.395
	AUC	**0.748**	**0.582**	0.697	0.555	0.705	0.509	0.667	0.551	0.594	0.537	0.499	0.497	0.709	0.558	0.561	0.547	0.550	0.568
Llama3	$F1$	0.211	0.447	0.156	0.406	0.170	0.405	0.147	0.413	**0.223**	**0.449**	0.184	0.421	0.184	0.422	0.158	0.417	0.208	0.414
	AUC	**0.773**	0.622	0.724	0.546	0.741	0.572	0.662	0.605	0.666	**0.626**	0.699	0.562	0.730	0.586	0.634	0.613	0.601	0.600
GPT4o mini	$F1$	**0.329**	**0.494**	0.185	0.395	0.183	0.395	0.250	0.395	0.279	0.487	0.277	0.488	0.085	0.395	0.135	0.395	0.085	0.395
	AUC	0.682	**0.668**	0.560	0.559	0.564	0.574	0.597	0.586	0.686	0.661	**0.703**	0.658	0.520	0.521	0.623	0.603	0.512	0.505
1st count		**16**	**12**	0	0	0	0	0	3	1	3	1	0	0	0	2	1	0	1

The consistency checks between edges are conducted iteratively until all candidate edges in $\tilde{\mathbb{E}}_i$ have been merged into the graph.

In summary, the iterative interaction between evaluation and elicitation processes continuously extends the knowledge hypergraph with intrinsic knowledge and retains the semantic relationships in the long-form context, facilitating subsequent evaluation and elicitation. Note that all knowledge in the graph is explicitly expressed by the models themselves, and no external information is included.

Output. After obtaining the hallucination score $\hat{p}_{i,j}$ for each statement $s_{i,j}$, we compute the prediction $\hat{y}_i$ for each candidate sentence r_i using maximum aggregation, i.e., $\hat{y}_i = \max(\hat{p}_{i,1}, \hat{p}_{i,2}, \dots)$, and similarly obtain $\hat{Y}$ for the full response R, i.e., $\hat{Y} = \max(\hat{y}_1, \hat{y}_2, \dots)$.

We provide detailed complexity analysis in Appendix B, implementation details in Appendix C.4, full prompts in Appendix F, and pseudo-code in Appendix G.

4 Experiments

4.1 Experimental Setups

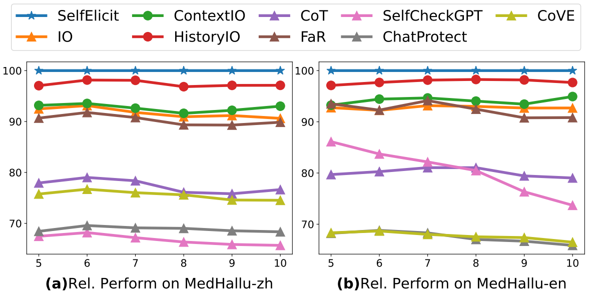
Datasets. We use the following long-form hallucination detection datasets: MedHallu-zh, MedHallu-en (medicine, see Appendix C.2 for the construction process), WikiBio (biography, Manakul et al., 2023), FActScore (biography, Min et al., 2023), and HaluEval2 (education, finance, and science, Li et al., 2024a). Table 2 shows the dataset statistics.

Table 2: Dataset Statistics. pos%: proportion of non-factuality. #Sent/Smp: number of sentences per sample.

Dataset	Split	Total (pos%)		#Sent/Smp		
		Response	Sentence	avg.	min	max
MedHallu-zh MedHallu-en	Train	1,622 (27.6%)	10,688 (5.3%)	6.59	1	22
	Validate	270 (24.1%)	1,809 (4.4%)	6.70	1	21
	Test	812 (24.6%)	5,534 (4.4%)	6.81	1	30
WikiBio	Validate	71 (71.8%)	571 (66.0%)	8.04	4	13
	Test	167 (75.4%)	1,337 (75.9%)	8.00	3	13
FActScore	Validate	138 (97.8%)	1,042 (57.3%)	7.55	4	16
	Test	323 (95.7%)	2,477 (54.2%)	7.67	4	15
HaluEval2	Validate	245 (89.4%)	1,035 (45.8%)	4.40	1	5
	Test	551 (90.9%)	2,464 (45.8%)	4.47	1	6

Models. We use the following off-the-shelf language models: Qwen1.5-7B-chat (Qwen, Bai et al., 2023), Qwen2.5-7B-Instruct (Qwen2, Bai et al., 2023), ChatGLM3-6B (GLM, GLM et al., 2024), Llama2-7B-chat (Llama2, Touvron et al., 2023), Llama3.1-8B-Instruct (Llama3, AI@Meta, 2024), and GPT4o-mini (OpenAI, 2024) for experiments. All language models use greedy decoding (temperature=0) during text generation for stable outputs.

Baselines. We compare our method with the following baselines, including classic calibration factchecking (IO, Kadavath et al., 2022; Mahaut et al., 2024), long-form enhanced methods (ContextIO and HistoryIO), and methods with various elicita-



(a) Rel. Perform on MedHallu-zh **(b)**Rel. Perform on MedHallu-en

Figure 3: Relative performance (%, average AUC v.s. SelfElicit) on samples with various lengths (numbers of sentences) with Qwen. x-axis: samples with length ≥x.

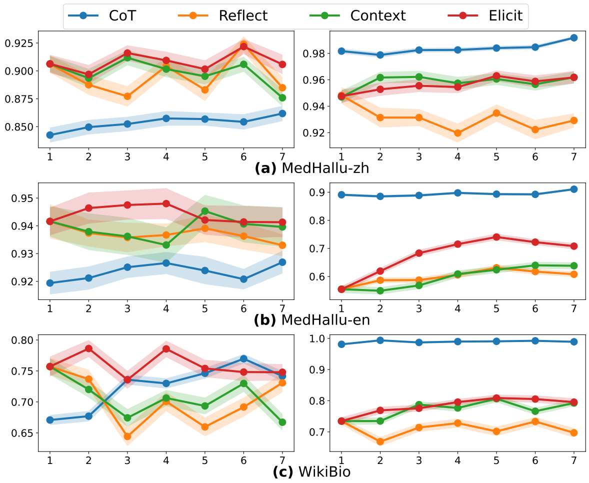
tion approaches: chain-of-thought (CoT, Wei et al. 2022 and FaR, Zhao et al., 2024), self-ask (CoVE, Dhuliawala et al., 2024), and self-consistency (SelfCheckGPT, Manakul et al., 2023, and ChatProtect, Mündler et al., 2024). For all methods, we use an identical prompt after their original procedures to obtain the hallucination score for a fair comparison (i.e., only elicitation approaches are different). More details are shown in Appendix C.1.

Metrics. We treat the hallucination detection as a classification task, where positive labels refer to non-factual statements. $F1$ and $AUROC$ are used for both sentence-wise and response-wise metrics. Since the threshold variance affects the metrics (Huang et al., 2024), we search for the best thresholds with the highest sentence/response $F1$ metrics independently on the validation set and regard hallucination scores larger than the thresholds as positive predictions on the test set.

4.2 Main Results

Table 1 shows the overall detection results. Appendix E shows supplementary results with specific domains on HaluEval2 and severities on MedHallu.

Multi-step elicitation reasoning generally shows compromised performance. Methods requiring multi-step reasoning (e.g., CoVE and ChatProtect) generally have inferior performance compared with other methods, while methods with straightforward prompts (e.g., IO) to utilize the models' knowledge generally perform better. Drawing from the conclusion of previous research (Sprague et al., 2024), the primary benefit of multi-step reasoning comes in the ability to execute symbolic steps and track the outputs (Sprague et al., 2024), rather than direct knowledge assessment. On the contrary, the risk of inaccuracies (e.g., information missing when generating questions in CoVE and triple ambiguity in ChatProtect) accumulates as the reasoning steps increase, potentially harming their overall capabilities to fully utilize the models' intrinsic knowledge.



(a) MedHallu-zh

(b) MedHallu-en

(c) WikiBio

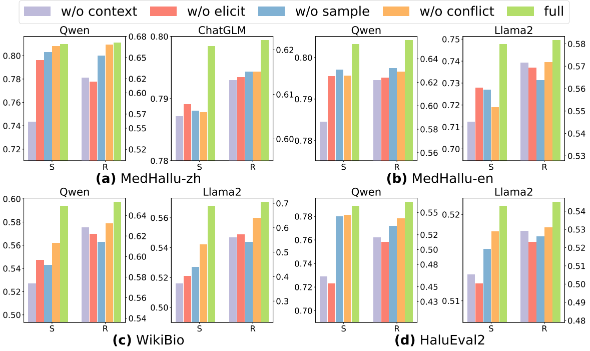
Figure 4: Factuality (left) and diversity (right) of elicited knowledge with different elicitation methods. x-axis refers to the statement's index number.

Long-form understanding benefits the detection. In Figure 3, methods without context (e.g., IO and SelfCheckGPT) show inferior performance compared with context-enhanced methods (e.g., ContextIO and HistoryIO). As the length of the sample increases, the gap between these two categories increases since the contextual information facilitates the understanding of statements and reasoning for fact-checking.

SelfElicit generally achieves superior performance across LLMs. Comparing results across various families of modern LLMs, SelfElicit consistently outperforms baselines (ranking top tiers in 37/40 cases). Conceptually, thoughts from prior statements ease the fact-check reasoning with in-context knowledge (Wang et al., 2022; Lee et al., 2023a) and also provide coherent understandings of the logical relationships across sentences. The self-consistency checking mechanism also helps to alleviate the snowballing of hallucinated content.

4.3 Elicitation Quality

We take a closer look into the elicitation by comparing the factuality (whether the reflection is factual) and diversity (whether the reflection is not identical to the statement) of the elicited knowledge with different elicitating methods: (1) *CoT*: chain-of-thought expression, (2) *Reflect*: calibration-guided reflection, (3) *Context*: reflection with prior statements, and (4) *Elicit*: reflection with previously generated thoughts. The reflections are generated



(a) MedHallu-zh (b) MedHallu-en (c) WikiBio (d) HaluEval2

Figure 5: Ablation results (AUC).

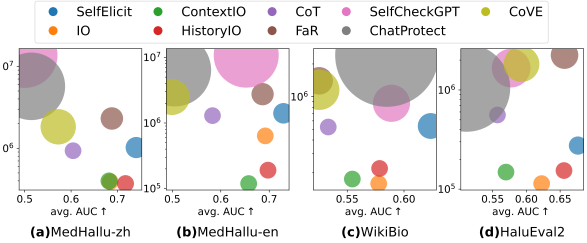
by Qwen and assessed by GPT4. We assign an index for each sentence within a multi-sentence response (starting from 1) and report the results averaged on the index in Figure 4.

We observe that (1) *CoT* has the lowest factuality rates and highest diversity, indicating a higher risk of induced hallucinations. On the contrary, evaluating before reflection can guide the elicitation with models' calibration and alleviate fabrications. (2) Prefixing context (*Context*) improves both the factuality and diversity in most cases (vs *Reflect*), suggesting that leveraging the contextual relations benefits the knowledge understanding. (3) Combined with conflict resolution, *Elicit* can consistently improve the expression of intrinsic knowledge with better factuality and diversity.

4.4 Ablation Study

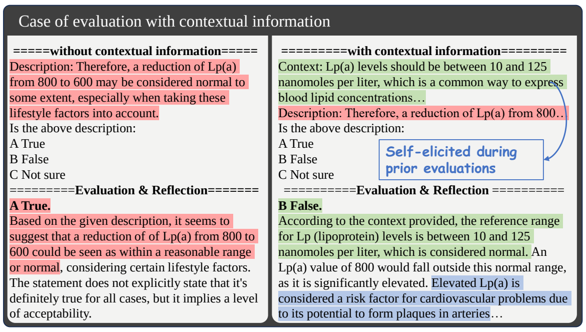
We conduct an ablation study to demonstrate the effectiveness of each component of our method. The variants include: (1) *w/o context*: fact-checking without sampled contextual knowledge, (2) *w/o elicit*: using prior statements rather than reflections as context, (3) *w/o sample*: linearizing the entire graph rather than sampling relevant knowledge as context, (4) *w/o conflict*: merging all new edges without inconsistencies mitigation, and (5) *full*: full SelfElicit method with all components. Figure 5 and Appendix E.3 show the ablation results.

We conclude that: (1) *w/o context* shows a salient degradation compared with other variants, highlighting the importance of contextual semantics. (2) Variants without sampling or conflict mitigation provide relatively inferior performance compared to *full*, even the worst in several cases (e.g., subfigure(d)). Manual reviews show that providing irrelevant or self-contradictory context to models will largely disturb their focus and affect their reasoning, demonstrating the importance of sampling and conflict mitigation components. (3) Versus all



(a)MedHallu-zh (b)MedHallu-en (c)WikiBio (d)HaluEval2

Figure 6: Inference costs with Qwen. x-axis: average AUC metrics. y-axis: number of output tokens (log). Spot size refers to the number of model calls.


Case of evaluation with contextual information

=====without contextual information=====

Description: Therefore, a reduction of Lp(a) from 800 to 600 may be considered normal to some extent, especially when taking these lifestyle factors into account.

Is the above description:

A True

B False

C Not sure

=========Evaluation & Reflection=======

A True.

Based on the given description, it seems to suggest that a reduction of of Lp(a) from 800 to 600 could be seen as within a reasonable range or normal, considering certain lifestyle factors. The statement does not explicitly state that it's definitely true for all cases, but it implies a level of acceptability.

=========with contextual information=========

Context: Lp(a) levels should be between 10 and 125 nanomoles per liter, which is a common way to express blood lipid concentrations…

Description: Therefore, a reduction of Lp(a) from 800…

Is the above description:

A True

B False

C Not sure

==========Evaluation & Reflection ==========

B False.

According to the context provided, the reference range for Lp (lipoprotein) levels is between 10 and 125 nanomoles per liter, which is considered normal. An Lp(a) value of 800 would fall outside this normal range, as it is significantly elevated. Elevated Lp(a) is considered a risk factor for cardiovascular problems due to its potential to form plaques in arteries…

Figure 7: A comparison of evaluation and reflection without(left) or with(right) contextual information. Red : non-factual content. Green : factual content. Blue : newly elicited content.

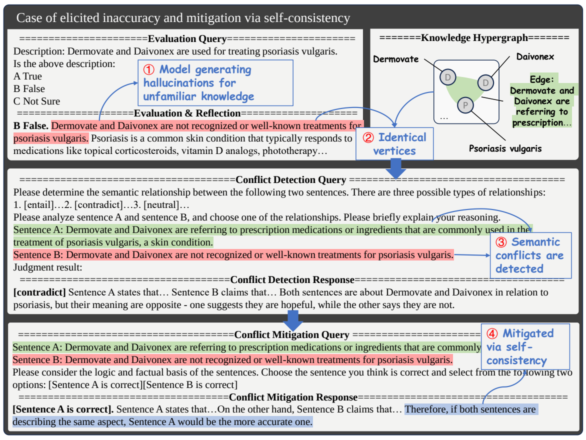
ablated variants, the full method generally provides the best performance. The full method outperforms 6.8% over *w/o context*, 6.0% over *w/o elicit*, 5.1% over *w/o sample*, and 2.9% over *w/o conflict* on average, highlighting the synergistic effect of all constituted parts.

4.5 Inference Costs

Figure 6 shows the inference costs for all methods. SelfElicit achieves the best performance with moderate costs in both the number of model calls and output tokens. Compared with multi-step reasoning methods having lower efficiency (e.g., SelfCheckGPT, ChatProtect, and CoVE), our method shows that simple reflections can also be effective in eliciting intrinsic knowledge. Specifically, SelfElicit has a comparable cost to chain-of-thought and consistently outperforms. Detailed metrics are in Appendix E.4. Theoretical complexity and scalability analysis are in Appendix B.

4.6 Case Study

We show a case of evaluation with/without contextual information in Figure 7. On the left side, the model has difficulty evaluating the statement since it fails to recall the normal range of Lp(a) or compare the values in a single reasoning step. On the contrary, on the right side, thoughts elicited



Case of elicited inaccuracy and mitigation via self-consistency

======================Evaluation Query======================

Description: Dermovate and Daivonex are used for treating psoriasis vulgaris.
Is the above description:
A True
B False
C Not Sure

==================Evaluation & Reflection==================

B False. Dermovate and Daivonex are not recognized or well-known treatments for psoriasis vulgaris. Psoriasis is a common skin condition that typically responds to medications like topical corticosteroids, vitamin D analogs, phototherapy…

=======Knowledge Hypergraph=======

==================================Conflict Detection Query ==================================

Please determine the semantic relationship between the following two sentences. There are three possible types of relationships:
1. [entail]…2. [contradict]…3. [neutral]…
Please analyze sentence A and sentence B, and choose one of the relationships. Please briefly explain your reasoning.
Sentence A: Dermovate and Daivonex are referring to prescription medications or ingredients that are commonly used in the treatment of psoriasis vulgaris, a skin condition.
Sentence B: Dermovate and Daivonex are not recognized or well-known treatments for psoriasis vulgaris.
Judgment result:

==================================Conflict Detection Response==================================

[contradict] Sentence A states that… Sentence B claims that… Both sentences are about Dermovate and Daivonex in relation to psoriasis, but their meaning are opposite - one suggests they are hopeful, while the other says they are not.

==================================Conflict Mitigation Query ==================================

Sentence A: Dermovate and Daivonex are referring to prescription medications or ingredients that are commonly
Sentence B: Dermovate and Daivonex are not recognized or well-known treatments for psoriasis vulgaris.
Please consider the logic and factual basis of the sentences. Choose the sentence you think is correct and select from the following two options: [Sentence A is correct][Sentence B is correct]

==================================Conflict Mitigation Response==================================

[Sentence A is correct]. Sentence A states that…On the other hand, Sentence B claims that… Therefore, if both sentences are describing the same aspect, Sentence A would be the more accurate one.

① Model generating hallucinations for unfamiliar knowledge
② Identical vertices
③ Semantic conflicts are detected
④ Mitigated via self-consistency
Dermovate
Daivonex
D
D
P
…
Edge: Dermovate and Daivonex are referring to prescription…
Psoriasis vulgaris

Figure 8: A case of generating inaccurate reflection and how it is mitigated by conflict detection module. Red : non-factual content. Green : factual content.

from prior statements provide direct information (the normal range of Lp(a)) to facilitate the evaluation of the current statement. Moreover, a new piece of knowledge about the indication of Lp(a) (blue) is elicited along with the evaluation.

Figure 8 shows a case when the model generates inaccurate reflection due to unfamiliarity with specific knowledge and how the inaccuracy is mitigated by conflict detection. Since Dermovate and Daivonex are trade names that are less exposed than their pharmaceutical names[1], the model ①fails to fact-check the statement and generates erroneous reflections (red). Such generated hallucinations might accumulate and finally affect the reasoning of subsequent evaluations. Thanks to the ②conflict detection module, i.e., the erroneous reflection shares an identical vertex set (Dermovate, Daivonex, and Psoriasis) with existing edges in the hypergraph, ③the NLI component is activated and identifies their semantic contradiction. Finally, ④the conflict is mitigated via self-consistency to avoid the snowballing of errors.

5 Disscussion

5.1 Connection with RAG

SelfElicit and retrieval-argument generations (RAG) (Jin et al., 2024; Luo et al., 2024; Sun et al., 2024) share some similarities in their schemas, i.e., retrieving relative information from a knowledge graph to facilitate the down-streaming tasks. Recent works (Sansford et al., 2024; Yuan et al., 2024; Niu et al., 2024) have demonstrated the performance gain to incorporate external knowledge graphs for hallucination detection. Differently, our work organizes knowledge graphs elicited from the models themselves, rather than relying on external databases. Moreover, compared with RAG methods where databases are stand-alone, the self-elicited knowledge hypergraphs in our framework depend on the models and evolve in parallel with the evaluation processes. Theoretically, our method is orthogonal to these RAG methods and can be integrated with these methods into a unified design, which might further benefit both the elicitation and hallucination detection.

5.2 Comparison with chain reasoning methods

Compared with chain reasoning methods, our method differs in three aspects. **Knowledge Management.** Our method utilizes a structured graph as the knowledge base, enabling multi-path reasoning and dynamic updates, whereas chain reasoning methods rely on hidden states and linear text expressions. Our method also supports explicit knowledge verification, editing, and external knowledge integration in future works. **Conflict Resolution.** Our method employs a self-consistency-based mechanism to resolve conflicts through NLI actively, whereas, in chain reasoning methods, unsolved inaccuracies may disturb the generation and lead to self-contradictions. **Scalability.** Through iterations and knowledge sampling, our method is preferred for long-form hallucination detection, while chain reasoning methods might suffer from excessively long inputs and reduced reasoning capabilities, leading to confused output format or hallucinations.

5.3 Domain-specific Adaption

Experiments in various domains (medicine, biography, finance, science, and education) have shown the generalizability of our method. Our method can also be applied to a wider range of domains. The only adaption required is the prompt of extracting entities/statements (Section 3.1), which is known as open information extraction (Niklaus et al., 2018). Incorporating domain-specific expertise will improve the quality of this procedure. All other prompts used in our framework are domain-agnostic and no additional adaptation is required.

6 Conclusion

In this paper, we have investigated the task of detecting hallucinations from long-form content. Ex-

[1] Manual tests show the model is less familiar with them.

isting methods predominantly fall short of comprehensively elicitating the intrinsic knowledge of models or overlook the semantic relationships within long-form content. To address these issues, we present a novel framework, SelfElicit, which uses self-generated thoughts from prior statements to elicit the models' intrinsic knowledge and synergize self-elicitation and contextual understanding in a unified diagram. Extensive experiments on five datasets with various domains with modern large language models have shown the effectiveness and superiority of the proposed framework.

Limitations

Some of the limitations of this work include: (1) this work primarily focuses on technological methods to elicit the intrinsic knowledge of models, leaving the question of why self-elicitation, as a mechanism, fundamentally improves knowledge elicitation to future works. Understanding whether LLMs either abstract knowledge over linguistic forms or merely memorize statements (Carlini et al., 2022) could provide more insights into hallucinations. (2) Since SelfElicit functions in an iterative loop, inaccuracies might gradually accumulate and the complexities increase as the length of response R increases, which might limit the application for large-scale deployments. (3) Besides factual hallucinations in long-form content, evaluating additional metrics, such as contextual appropriateness and logical coherence will enhance the assessment of generated content. (4) The performance upper bound is theoretically restricted by the models' capacity obtained from pre-training, leaving continual improvements an open question.

Ethical Considerations

The statements and examples provided in this paper are intended for demonstration purposes only and may contain non-factual information. Our intent is to illustrate concepts rather than present verified facts. Readers are strongly advised to consult with professional practitioners or academic experts before taking any actions in high-risk scenarios. The MedHallu datasets are collected and annotated by third-party certified organizations ethically and legally. All datasets in this paper are anonymized and intended for research only.

Acknowledgments

This work was supported by the National Key Research and Development Program of China (2022YFB4500300), and the National Natural Science Foundation of China (62202422 and 62372408). We extend our gratitude to Zelin Dai and Xinxin Lu from Alibaba Health for their insightful discussions on the project. We also thank the technical team at Alibaba Health for their invaluable support.

References

AI@Meta. 2024. Llama 3 model card.

Amos Azaria and Tom Mitchell. 2023. The Internal State of an LLM Knows When It's Lying. *Preprint*, arXiv:2304.13734.

Jinze Bai, Shuai Bai, et al. 2023. Qwen Technical Report. *Preprint*, arXiv:2309.16609.

Yejin Bang, Samuel Cahyawijaya, et al. 2023. A Multitask, Multilingual, Multimodal Evaluation of ChatGPT on Reasoning, Hallucination, and Interactivity. In *AACL-IJCNLP*.

Maciej Besta, Nils Blach, et al. 2024. Graph of Thoughts: Solving Elaborate Problems with Large Language Models. In *AAAI*, volume 38.

Nicholas Carlini, Daphne Ippolito, et al. 2022. Quantifying Memorization Across Neural Language Models. In *ICLR*.

Xiusi Chen, Jyun-Yu Jiang, et al. 2024. MinPrompt: Graph-based Minimal Prompt Data Augmentation for Few-shot Question Answering. In *ACL*.

Yung-Sung Chuang, Yujia Xie, et al. 2024. DoLa: Decoding by Contrasting Layers Improves Factuality in Large Language Models. In *ICLR*.

Roi Cohen, May Hamri, et al. 2023. LM vs LM: Detecting Factual Errors via Cross Examination. In *EMNLP*.

Preetam Prabhu Srikar Dammu, Himanshu Naidu, et al. 2023. Claimver: Explainable claim-level verification and evidence attribution of text through knowledge graphs. In *EMNLP*.

Zhenyun Deng, Michael Schlichtkrull, et al. 2024. Document-level Claim Extraction and Decontextualisation for Fact-Checking. In *ACL*.

Shehzaad Dhuliawala, Mojtaba Komeili, et al. 2024. Chain-of-Verification Reduces Hallucination in Large Language Models. In *ACL Findings*.

Darren Edge, Ha Trinh, et al. 2024. From Local to Global: A Graph RAG Approach to Query-Focused Summarization. *Preprint*, arXiv:2404.16130.

Ekaterina Fadeeva, Aleksandr Rubashevskii, et al. 2024. Fact-Checking the Output of Large Language Models via Token-Level Uncertainty Quantification. In *ACL Findings*.

Sebastian Farquhar, Jannik Kossen, et al. 2024. Detecting hallucinations in large language models using semantic entropy. *Nature*, 630(8017).

Team GLM, Aohan Zeng, et al. 2024. ChatGLM: A Family of Large Language Models from GLM-130B to GLM-4 All Tools. *Preprint*, arXiv:2406.12793.

Zhibin Gou, Zhihong Shao, et al. 2024. CRITIC: Large Language Models Can Self-Correct with Tool-Interactive Critiquing. In *ICLR*.

Jian Guan, Jesse Dodge, et al. 2024. Language Models Hallucinate, but May Excel at Fact Verification. In *NAACL*.

Lei Huang, Weijiang Yu, et al. 2023. A Survey on Hallucination in Large Language Models: Principles, Taxonomy, Challenges, and Open Questions. *Preprint*, arXiv:2311.05232.

Xinmeng Huang, Shuo Li, et al. 2024. Uncertainty in Language Models: Assessment through Rank-Calibration. *Preprint*, arXiv:2404.03163.

Jinhao Jiang, Kun Zhou, et al. 2023. StructGPT: A General Framework for Large Language Model to Reason over Structured Data. In *EMNLP*.

Bowen Jin, Chulin Xie, et al. 2024. Graph Chain-of-Thought: Augmenting Large Language Models by Reasoning on Graphs. *Preprint*, arXiv:2404.07103.

Saurav Kadavath, Tom Conerly, et al. 2022. Language Models (Mostly) Know What They Know. *Preprint*, arXiv:2207.05221.

Ryo Kamoi, Yusen Zhang, et al. 2024. When Can LLMs Actually Correct Their Own Mistakes? A Critical Survey of Self-Correction of LLMs. *Preprint*, arXiv:2406.01297.

Haoqiang Kang, Juntong Ni, and Huaxiu Yao. 2023. Ever: Mitigating Hallucination in Large Language Models through Real-Time Verification and Rectification. *Preprint*, arXiv:2311.09114.

Lorenz Kuhn, Yarin Gal, and Sebastian Farquhar. 2023. Semantic Uncertainty: Linguistic Invariances for Uncertainty Estimation in Natural Language Generation. In *ICLR*.

Gibbeum Lee, Volker Hartmann, et al. 2023a. Prompted LLMs as Chatbot Modules for Long Open-domain Conversation. In *ACL*.

Nayeon Lee, Wei Ping, et al. 2023b. Factuality Enhanced Language Models for Open-Ended Text Generation. In *NeurIPS*.

Junyi Li, Jie Chen, Ruiyang Ren, et al. 2024a. The dawn after the dark: An empirical study on factuality hallucination in large language models. *arXiv preprint arXiv:2401.03205*.

Kenneth Li, Oam Patel, et al. 2023a. Inference-Time Intervention: Eliciting Truthful Answers from a Language Model. In *NeurIPS*.

Miaoran Li, Baolin Peng, et al. 2024b. Self-Checker: Plug-and-Play Modules for Fact-Checking with Large Language Models. In *NAACL*.

Weitao Li, Junkai Li, et al. 2024c. Citation-Enhanced Generation for LLM-based Chatbots. *Preprint*, arXiv:2402.16063.

Xiaonan Li, Changtai Zhu, et al. 2023b. LLatrieval: LLM-Verified Retrieval for Verifiable Generation. In *NAACL*.

Linhao Luo, Yuan-Fang Li, et al. 2024. Reasoning on Graphs: Faithful and Interpretable Large Language Model Reasoning. In *ICLR*.

Matéo Mahaut, Laura Aina, et al. 2024. Factual Confidence of LLMs: On Reliability and Robustness of Current Estimators. In *ACL*.

Potsawee Manakul, Adian Liusie, and Mark J. F. Gales. 2023. SelfCheckGPT: Zero-Resource Black-Box Hallucination Detection for Generative Large Language Models. In *EMNLP*.

Ning Miao, Yee Whye Teh, and Tom Rainforth. 2024. SelfCheck: Using LLMs to Zero-Shot Check Their Own Step-by-Step Reasoning. In *ICLR*.

Sewon Min, Kalpesh Krishna, et al. 2023. FActScore: Fine-grained Atomic Evaluation of Factual Precision in Long Form Text Generation. In *EMNLP*.

Abhika Mishra, Akari Asai, et al. 2024. Fine-grained Hallucination Detection and Editing for Language Models. *Preprint*, arXiv:2401.06855.

Niels Mündler, Jingxuan He, et al. 2024. Self-contradictory Hallucinations of Large Language Models: Evaluation, Detection and Mitigation. In *ICLR*.

NHS. 2024. How and when to take gliclazide. https://www.nhs.uk/medicines/gliclazide/how-and-when-to-take-gliclazide/.

Christina Niklaus, Matthias Cetto, André Freitas, and Siegfried Handschuh. 2018. A survey on open information extraction. *arXiv preprint arXiv:1806.05599*.

Mengjia Niu, Hao Li, et al. 2024. Mitigating Hallucinations in Large Language Models via Self-Refinement-Enhanced Knowledge Retrieval. *Preprint*, arXiv:2405.06545.

OpenAI. 2024. Gpt-4o system card. *Preprint*, arXiv:2410.21276.

Fabio Petroni, Tim Rocktäschel, et al. 2019. Language Models as Knowledge Bases? In *EMNLP-IJCNLP*.

Shanghaoran Quan, Tianyi Tang, Bowen Yu, et al. 2024. Language models can self-lengthen to generate long texts. *Preprint*, arXiv:2410.23933.

Haoran Que, Feiyu Duan, Liqun He, et al. 2024. Hellobench: Evaluating long text generation capabilities of large language models. *Preprint*, arXiv:2409.16191.

Hannah Sansford, Nicholas Richardson, et al. 2024. GraphEval: A Knowledge-Graph Based LLM Hallucination Evaluation Framework. *Preprint*, arXiv:2407.10793.

Ritvik Setty and Vinay Setty. 2024. QuestGen: Effectiveness of Question Generation Methods for Fact-Checking Applications. In *CIKM*.

Zayne Sprague, Fangcong Yin, et al. 2024. To CoT or not to CoT? Chain-of-thought helps mainly on math and symbolic reasoning. *Preprint*, arXiv:2409.12183.

Jiashuo Sun, Chengjin Xu, et al. 2024. Think-on-Graph: Deep and Responsible Reasoning of Large Language Model on Knowledge Graph. In *ICLR*.

Shuchang Tao, Liuyi Yao, et al. 2024. When to Trust LLMs: Aligning Confidence with Response Quality. In *ACL Findings*.

Katherine Tian, Eric Mitchell, et al. 2024. Fine-tuning Language Models for Factuality. In *ICLR*.

Hugo Touvron, Louis Martin, et al. 2023. Llama 2: Open Foundation and Fine-Tuned Chat Models. *Preprint*, arXiv:2307.09288.

Boshi Wang, Xiang Deng, et al. 2022. Iteratively Prompt Pre-trained Language Models for Chain of Thought. In *EMNLP*.

Huazheng Wang, Haifeng Sun, et al. 2024a. SSS: Editing Factual Knowledge in Language Models towards Semantic Sparse Space. In *ACL Findings*.

Jianing Wang, Qiushi Sun, et al. 2024b. Boosting Language Models Reasoning with Chain-of-Knowledge Prompting. In *ACL*.

Jason Wei, Xuezhi Wang, et al. 2022. Chain-of-Thought Prompting Elicits Reasoning in Large Language Models. In *NeurIPS*, volume 35.

Jerry Wei, Chengrun Yang, et al. 2024. Long-form factuality in large language models. *Preprint*, arXiv:2403.18802.

Orion Weller, Marc Marone, et al. 2024. "According to ...": Prompting Language Models Improves Quoting from Pre-Training Data. In *EACL*.

Thomas Wolf, Lysandre Debut, et al. 2020. Transformers: State-of-the-Art Natural Language Processing. In *EMNLP*.

Yuan Xia, Jingbo Zhou, et al. 2024. Improving Retrieval Augmented Language Model with Self-Reasoning. *Preprint*, arXiv:2407.19813.

Yakir Yehuda, Itzik Malkiel, et al. 2024. InterrogateLLM: Zero-Resource Hallucination Detection in LLM-Generated Answers. In *ACL*.

Wenhao Yu, Zhihan Zhang, et al. 2023. Improving Language Models via Plug-and-Play Retrieval Feedback. *Preprint*, arXiv:2305.14002.

Moy Yuan, Andreas Vlachos, et al. 2024. Zero-Shot Fact-Checking with Semantic Triples and Knowledge Graphs. In *KaLLM*.

Zhenrui Yue, Huimin Zeng, et al. 2024. Retrieval Augmented Fact Verification by Synthesizing Contrastive Arguments. In *ACL*.

Muru Zhang, Ofir Press, et al. 2023a. How Language Model Hallucinations Can Snowball. *Preprint*, arXiv:2305.13534.

Shaolei Zhang, Tian Yu, et al. 2024a. TruthX: Alleviating Hallucinations by Editing Large Language Models in Truthful Space. In *ACL*.

Tianhang Zhang, Lin Qiu, et al. 2023b. Enhancing Uncertainty-Based Hallucination Detection with Stronger Focus. In *EMNLP*.

Tianyi Zhang, Varsha Kishore, et al. 2020. BERTScore: Evaluating Text Generation with BERT. In *ICLR*.

Xiaokang Zhang, Zijun Yao, et al. 2024b. Transferable and Efficient Non-Factual Content Detection via Probe Training with Offline Consistency Checking. In *ACL*.

Xiaoying Zhang, Baolin Peng, et al. 2024c. Self-Alignment for Factuality: Mitigating Hallucinations in LLMs via Self-Evaluation. In *ACL*.

Xinran Zhao, Hongming Zhang, et al. 2024. Fact-and-Reflection (FaR) Improves Confidence Calibration of Large Language Models. In *ACL Findings*.

A RelatedWorks

A.1 Hallucination Detection

Retrieval-argument methods. Extracting relevant knowledge from external authentic database and incorporating it with the query is a common way of detecting hallucination (Min et al., 2023; Tian et al., 2024; Gou et al., 2024; Li et al., 2024c; Xia et al., 2024). Li et al. (2023b); Yu et al. (2023); Wei et al. (2024) proposed to update the retrieval results with LLM until the retrieved documents adequately support answering the questions. Kamoi et al. (2024); Yuan et al. (2024); Sansford et al. (2024) extracted keywords as entities and knowledge as triples and retrieved reference triples from knowledge graphs or texts. Additionally, Yue et al. (2024) contrasted the supportive arguments and refuting arguments derived from retrieval evidence.

Probe-based methods. Probe-based methods aim to understand the hallucination within the hidden activations of deeper model layers (Azaria and Mitchell, 2023; Zhang et al., 2024b; Wang et al., 2024a). They usually required probes pre-trained on a specific dataset to detect the hallucinations (Li et al., 2023a; Zhang et al., 2024a).

Intrinsic Knowledge-based methods. We categorize existing knowledge-based hallucination detection methods into three types. (1) Some methods focus on the token probabilities of white-box LLMs. Kadavath et al. (2022); Tian et al. (2024) proposed a calibration-based method to evaluate the correctness of the content with multiple-choice questions. Extending the token entropy estimation (Manakul et al., 2023) with keyword focusing, Zhang et al. (2023b) proposed to penalize the attention score of the hallucinated token to avoid snowballing (Zhang et al., 2023a). FaR (Zhao et al., 2024) and CoK (Wang et al., 2024b) elicited the intrinsic knowledge relevant to the query and reflected on the knowledge to improve the calibration. (2) Some methods propose to ask LLMs to express their uncertainty verbally (Mahaut et al., 2024). Tao et al. (2024) leverages reinforcement learning guided by a tailored dual-component reward function. (3) Other methods aim at the semantic consistency over sentences (Kuhn et al., 2023; Manakul et al., 2023; Mündler et al., 2024; Miao et al., 2024). SelfCheckGPT (Manakul et al., 2023), Kuhn et al. (2023) and Farquhar et al. (2024) estimated the variance of the meaning of generated content. Cohen et al. (2023) discovered the inconsistencies with the interaction between LLMs. InterrogateLLM (Yehuda et al., 2024) reversed the query-response pair and estimated the variation of reconstructed queries for semantic uncertainty. ChatProtect (Mündler et al., 2024) and SelfCheck (Miao et al., 2024) detected hallucinations by comparing the original content and the regenerated one. EVER (Kang et al., 2023), CoVE (Dhuliawala et al., 2024), Zhang et al. (2024c), Farquhar et al. (2024), and QuestGen (Setty and Setty, 2024) generated questions corresponding to each fact within the content, answered the generated question, and measured the coherence between the answer and the original content.

Compared with the above works, our method uses self-generated thoughts as a catalyst to elicit intrinsic knowledge, without external databases, finetuning, or complex multi-step reasoning. Meanwhile, the iterative schema captures the contextual relationships of long-form content and the conflict mitigation mechanism reduces induced hallucination.

A.2 Large Language Models with Knowledge Graphs

Efforts have been made to facilitate large language models for reasoning or factuality with knowledge graphs. GoT (Besta et al., 2024) used a graph structure to guide the reasoning of LLMs. Yuan et al. (2024) proposed to extract knowledge graphs from external text databases and regarded fact-checking as a task of NLI. GraphRAG (Edge et al., 2024) built a graph-based text index by deriving entity knowledge graphs from the source documents and generating summaries for hierarchical graph communities. RoG (Luo et al., 2024) synergized LLMs reasoning with KGs to improve the ability of knowledge traceability and knowledge correctability. ToG (Sun et al., 2024) and Graph-CoT (Jin et al., 2024) treated the LLM as an agent to interactively explore related entities and relations on KGs and perform reasoning based on the retrieved knowledge. Re-KGR (Kamoi et al., 2024) and StructGPT (Jiang et al., 2023) leveraged knowledge graphs as external databases and directly retrieved reference information for factual QA. Sansford et al. (2024) converted the response into a candidate knowledge graph and fact-checked each triple in the graph. Compared with the above methods, our method does not rely on external knowledge graphs but uses self-elicited knowledge to construct the graph to facilitate hallucination detection.

Table 3: Complexity analysis of all methods. N: the number of sentences. F: the number of factoids. K: the number of generated documents. Brackets denote the components that require LLM usage.

Method	Token	Open-ended (major overhead)
IO	N (fact-check)	-
ContextIO	N (fact-check)	-
HistoryIO	N (fact-check)	N (reflect)
CoT	N (fact-check)	N (reason)
CoVE	FN (result compare)	N (generate questions) + FN (answer)
FaR	N (fact-check)	N(elicit) + N(reflect)
SelfCheckGPT	KN (fact-check)	K (generate document)

ChatProtect	FN(eval)	N(extract triple)
		+ FN(cloze triple)

SelfElicit	F (fact-check) + F(detect conflict) + F (mitigate conflict)	1 (extract) + F (reflect)

B Complexity and Scalability

Complexity

Assume there are N statements (or F factoids, where $N \sim F$) to be fact-checked in a given sample. We categorize the LLM usages into two categories: **token** where only the first token or its logit matters, and **open-ended** where LLMs generate full reasoning given an instruction. We summarize the theoretical complexity of all methods in Table 3.

Since an open-ended generation is magnitudes more costly than a token generation because the former usually generates hundreds of tokens while the latter generates only several tokens, the major overhead comes from the open-ended generations. In summary, SelfElicit (1+F) has a comparable complexity with CoT (N) and is faster than CoVE (N+FN), FaR (N+N), and ChatProtect (N+FN). The detailed experimental results are shown in Appendix E.4.

Scalability

We discuss in this section the scalability of SelfElicit concerning increasingly longer samples (i.e., the number of statements within a sample increases). We denote the number of statements (or factoids) in a sample as F. The complexities of the components are:

- **sampling & merging**. Rule-based, no LLM usage. Since the sampling is merely edge retrieval rather than graph traversal, the overhead is trivial.
- **evaluation & reflection**. $\mathcal{O}(F)$ LLM calls in total to evaluate all statements one by one.
- **conflict detection & mitigation**. In extreme cases where each new edge conflicts with the existing ones in the graph, the complexity is $\mathcal{O}(F^2)$.

In summary, the overall complexity is $\mathcal{O}(F^2)$ for each input sample, which indicates an underlying quadratic increase of overhead if the number of factoids of a sample as F increases.

However, we argue that this scalability issue is not the major challenge in current hallucination detection practices. Specifically, the conflict detection/mitigation procedures are not always activated in all cases (e.g., the detection is activated in 17.76% (801/4510) cases, and the mitigation is activated in 1.06% (48/4510) cases with Qwen in MedHallu-en). More specifically, conflict detections contribute to 40k/1.4M (2.8%) token usages and mitigations contribute to 2.4k/1.4M (0.2%) token usages. Moreover, we optimize the implementation to limit the costs, such as early-stopping the generation during hypergraph updating (conflict detection and mitigation) processes since we only care about the first several tokens (e.g., [contradict] in conflict detections or [Sentence A is correct] in mitigations.

To handle potential dataset scalability challenges in the future, several in-place optimizations can be made, including (1) an improved graph sampling procedure to minimize conflict possibilities, (2) more efficient conflict detection approaches (e.g., embedding-based index), and (3) reducing the scale of the graph via clustering approaches.

C Experimental Details

C.1 Baselines

Our comparison includes representative methods that focus on retrieval-free, training-free methods for post-generation fact-checking, including classic calibration-based fact-checking:

- **IO** (Kadavath et al., 2022): Straightforwardly querying whether the statement is factual or not and obtaining the probability of `False` token as hallucination score.

context enhanced methods:

- **ContextIO**: Prior evaluated statements are prefixed as contextual information.

- **HistoryIO**: Historical information (both queries and responses) of prior evaluations are prefixed as contextual information.

and methods with various elicitation approaches:

- **CoT** (Wei et al., 2022): Prompting to evaluate the factuality of the given statement after step-by-step reasoning.
- **CoVE** (Dhuliawala et al., 2024): Generating verification questions given the statement, answering the questions independently, and summarizing for final evaluation. The factored answering strategy is employed. We manually craft 3 few-shot samples for question generation on each dataset.
- **FaR** (Zhao et al., 2024): Eliciting the knowledge relevant to the statement from models and asking models to reason on them to generate the final answer.
- **SelfCheckGPT** (Manakul et al., 2023): Querying to assess whether the statement is supported by stochastic documents answering the original user query. In the experiment, 5 stochastic documents are generated for each sample and the prompt-based setting is employed.
- **ChatProtect** (Mündler et al., 2024): Extracting knowledge triples, cloze triples, and predicting the contradiction between the given and the new statements. We manually craft 3 few-shot samples for triple extraction and cloze on each dataset.

We have excluded some related methods designed to quantify the factuality during generation rather than post-generation (Fadeeva et al., 2024; Zhang et al., 2023b; Yehuda et al., 2024) and methods required training on specific datasets (Zhang et al., 2024a; Wang et al., 2024a; Li et al., 2023a; Chuang et al., 2024; Wang et al., 2024b) or focusing on retrieval-argument approaches (Min et al., 2023; Tian et al., 2024; Li et al., 2024c; Xia et al., 2024). For all methods, we use an identical prompt (similar to IO) after their original procedures to obtain the hallucination score for a fair comparison, i.e., only elicitation approaches are different.

C.2 Datasets

MedHallu Datasets

We have collected a substantial dataset, namely MedHallu, by collecting genuine user queries and the corresponding responses generated by LLMs from an online healthcare QA platform. All identifying information is removed. This corpus mainly encompasses chronic diseases, cancer, and psoriasis and includes a Chinese version (with postfix `-zh`) and an English version (with postfix `-en`). The anonymous, query-response pairs are preprocessed with the following steps to obtain hallucination labels.

Step 1: Claim Parsing. The long-form response is first segmented into sentences by punctuation. Then, following (Wei et al., 2024), GPT-4 is used to split sentences into atomic claims, where each refers to a piece of information.

Step 2: Labeling. We ask certified medical experts to label whether each LLM-generated response includes any factual error or misunderstands the user query. Then, GPT4 is used to label each sentence given the response-wise human labels to obtain sentence-wise labels and claim-wise labels. We carefully check every positive sentence/claim such that they actually include factual errors.

Step 3: Multi-language. We use GPT4 to translate the Chinese QA pairs, sentences and claims into English. The prompt is constituted of the original user query and the candidate response and includes instructions to ask the LLM to take special care of the medical terminologies.

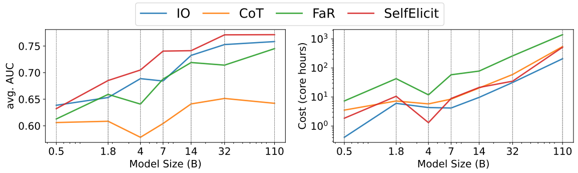
Step 4: Deduplication & Splitting. Samples with duplicated queries and responses are removed. Then all samples are randomly shuffled and split into train/validation/test sets with a ratio of 0.6/0.1/0.3. Table 10 shows a sample from MedHallu-en.

Other Datasets

For the WikiBio, FActScore, and HaluEval2 datasets, the split ratio is 0/0.3/0.7 and both minor and major inaccuracies are regarded as non-factual.

C.3 Data Preprocessing

As stated in previous works (Deng et al., 2024), the sentences might include information irrelevant to the central idea of the document. Verifying all information is inefficient and even misleading since some statements are simple repetitions of the user query or include subjective thoughts that are not directly relevant to the concept of factuality. To this end, we identify sentences that contain check-worthy statements, including assertions and thoughts regarding objective knowledge. Specifi-



(a) Model Size vs Performance **(b)**Model Size vs Cost

Figure 9: Performance and cost with different model scales. Log coordinates are used in both x-axes and y-axis in sub-figure(b).

cally, we provide the LLMs instructions and few-shot samples with domain-specific expertise and ask them to judge whether a sentence includes any objective knowledge. The selected check-worthy sentences are denoted as $\{r_1, r_2, \cdots\}$. For MedHallu-zh and MedHallu-en datasets, we use the prompt in Figure 14 for check-worthy statements. For other datasets, all statements are regarded as check-worthy.

C.4 Implementation Details

All experiments are conducted with transformers (Wolf et al., 2020) 4.43.0 on a Centos machine with Nvidia A800-80G GPUs. For statement extraction (Section 3.1), the extraction of entities and statements is achieved in a single chain-of-thought generation. For knowledge sampling (Section 3.2), we set $\alpha = 1$ and $\beta = 3$ practically. The generated order of the sampled statements is retained and duplicated sampled statements are removed. For evaluation and elicitation (Section 3.2), the logit of the first token of the output is used for the hallucination score, and other tokens are regarded as reflections. These reflections are split into sentences and filtered via manually crafted rules (e.g., discarding sentences with black-listed words) to obtain knowledgeable statements. For simplicity, we only obtain 1 reflection for each candidate statement. For knowledge storage and conflict resolution (Section 3.3), the order of the merged statements is kept so that newly generated statements are arranged after the older ones. All detailed prompts are shown in Appendix F.

D More Experiments

D.1 Model Scalability

We study the relationship between model scale and performance. We choose methods with preferable performance and efficiency (i.e., IO, CoT, FaR, and SelfElicit) for comparison and use the Qwen1.5-chat (Bai et al., 2023) family with model sizes 0.5B, 1.8B, 4B, 7B, 14B, 32B, and 110B.

Table 4: Results with different α-β pairs with Qwen on MedHallu-zh. **Bold**: the best. Underlined: the second best.

Match	α	β	sentence F1	sentence AUC	paragraph F1	paragraph AUC
strict	1	1	**0.272**	0.794	0.458	0.656
	1	2	0.265	**0.815**	0.452	0.651
	1	3	0.269	0.810	**0.475**	**0.671**
	3	3	0.242	0.783	0.434	0.611
relax	1	1	0.237	0.735	**0.461**	0.635
	1	2	**0.264**	**0.816**	0.453	**0.655**
	1	3	**0.264**	0.814	0.452	0.651
	3	3	0.255	0.760	0.444	0.622

The scaling of the performance and cost is shown in Figure 9. We have the following observations. (1) The trend generally follows the scaling law that larger models tend to have better performance and the inference costs also increase nearly linearly with the model size. (2) We notice a salient performance and cost degradation of the 4B model and a slightly higher cost for the 1.8B model. After manually checking the output, we found that the average output length of the 1.8B models is much longer than that of the 4B model. We believe the reason is the models' preference obtained during pre-training, rather than caused by hallucination detection approaches. (3) Both the 7B and 14B models achieve a good balance between performance and cost. Therefore, we choose the 7B model or models with similar scales to conduct all experiments in this paper. (4) Comparing all baselines, our SelfElicit almost achieves the best performance with all model scales, while having relatively similar inference with CoT.

D.2 Hyperparameter Sensitivity

By changing the α and β hyper-parameters in Equation 3, we can change the sampling scope from the knowledge hypergraph. We conduct experiments to investigate the choices of these hyper-parameters, and matching strategy. Matching strategy `strict` refers to sample an edge iff the query $\mathbb{V}_i(k)$ exactly match the vertex set of an edge, i.e., $e.\text{nodes} == \mathbb{V}_i(k)$. The `relax` refers to sample an edge if the query $\mathbb{V}_i(k)$ is a subset of the vertices of an edge, i.e., $e.\text{vertices} \in \mathbb{V}_i(k)$, providing a wider sampling scope.

Table 4 shows the result with different α-β pairs. We set the maximum value of both hyperparameters to 3 practically, since we found that combinations of more than 3 entities rarely sample

Table 5: Comparison of prompt and NLI model for semantic relationship prediction. **Bold**: the better one.

Dataset	LLM	Method	Sentence-wise		Response-wise	
			F1	AUC	F1	AUC
MedHallu-zh	Qwen	LLM prompt	0.269	**0.810**	**0.475**	**0.671**
		NLI model	0.269	0.794	0.469	0.664
	GLM	LLM prompt	**0.228**	**0.798**	0.445	0.622
		NLI model	0.227	0.793	**0.452**	**0.625**
MedHallu-en	Qwen	LLM prompt	**0.242**	**0.803**	**0.463**	**0.656**
		NLI model	0.237	0.789	0.455	0.647
	Llama2	LLM prompt	**0.181**	**0.748**	**0.408**	**0.582**
		NLI model	0.179	0.746	0.397	0.572

any edges. It can be observed that the performance is sensitive to the knowledge context sampled from the graph. A conservative sampling strategy ($\alpha = 1$, $\beta = 1$) will limit the utility of the knowledge in the graph, resulting in a performance closer to baselines IO (see Table 1). On the contrary, an excessively unrestricted sampling ($\alpha = 3$, $\beta = 3$) will result in more irrelevant information and longer input contexts, thereby limiting the performance. Therefore, we practically set $\alpha = 1$ and $\beta = 3$ in all other experiments for convenience.

D.3 NLI Method

We compare two different methods to predict the semantic relationship between two statements having identical entities (Equation 8): LLM prompts or specific pre-trained NLI models. For LLM prompts, we use prompt shown in Figure 12 and for NLI models, we use StructBERT[2] for MedHallu-zh and DeBERTa[3] for MedHallu-en. The results are listed in Table 5. It can be observed that using prompts consistently performs better than using specific NLI models. However, the differences are trivial and therefore we decided to use prompts in our implementation for convenience.

E Supplementary Results

E.1 Performance in each Domain on HaluEval2

HaluEval2 includes 197 education samples, 199 science samples, and 199 finance samples. The hallucination detection results in each domain are shown in Table 6. Regarding specific domains, our method achieves leading (best or second best) performance in 7 over 12 cases and SOTA in education/science domains with the Qwen model. These results show that SelfElicit has a strong performance across various domains. However, it is observed that SelfElicit does not achieve state-of-the-art in several cases, because the logical relationships among sentences are less correlated on datasets not specifically designed for long-form hallucination detection. As a result, the advantages of understanding contextual relationships are limited, restricting the performance gains of all context-argument methods (e.g., ContextIO, HistoryIO, and SelfElicit).

E.2 Performance with Different Severity on MedHallu-en

Distinguishing between high-risk and low-risk errors is essential in real-world clinical settings. To quantify the performance with different severity in medical datasets, we have asked the clinical expert to annotate all the erroneous samples in the test set of the MedHallu-en into 50 high-risk samples (e.g., duration, side effect, medication efficacy errors), 89 medium-risk samples (e.g., normal range, symptom errors), and 61 low-risk samples (e.g., health supplement efficacy, pill weight errors). The results are shown in Table 7.

We observe that SelfElicit outperforms all baselines in detecting hallucinations with various severities. Specifically, SelfElicit detects 86% high-risk errors (+5 samples, +10% than the best baseline), 87% medium-risk errors (+6 samples, +6% than the best baseline), and 92% low-risk errors (+2 samples, +3% than the best baseline). Moreover, despite ChatProtect has leading performance in detecting positive samples, it suffers from the most false positives. Among all methods, SelfElicit has the least false positives (8 samples less than the best baseline) and the highest precision (+0.11, +50% than the best baseline). These additional results have shown the significant leading performance of SelfElicit against baselines across different severity in medical domains.

E.3 Detailed Results of Ablation Study

Table 8 shows the detailed ablation results shown in Section 4.4.

E.4 Detailed Results of Inference Costs

Table 9 shows the detailed results on inference costs shown in Section 4.5.

F Prompts

Figure 10, Figure 12, Figure 13 show the prompts of SelfElicit.

[2] https://modelscope.cn/models/iic/nlp_structbert_nli_chinese-large

[3] https://huggingface.co/microsoft/deberta-large-mnli

Table 6: Hallucination detection results (AUC) for each domain (education, science, and finance) on HaluEval2 dataset with Qwen. S: sentence-level metrics. R: response-level metrics. **Red**: the best. Blue: the second best. Higher metrics are better.

LLM	**SelfElicit**		IO		ContextIO		HistoryIO		CoT		CoVE		FaR		SelfChkGPT		ChatProtect	
	S	R	S	R	S	R	S	R	S	R	S	R	S	R	S	R	S	R
HaluEval2-education																		
Qwen	**0.791**	**0.758**	0.709	0.651	0.708	0.593	0.762	0.711	0.618	0.604	0.665	0.499	0.702	0.481	0.609	0.562	0.521	0.490
Llama2	0.583	0.462	0.517	0.448	0.503	0.467	0.495	0.484	**0.613**	0.591	0.549	0.575	0.556	**0.613**	0.556	0.501	0.542	0.601
HaluEval2-science																		
Qwen	**0.831**	0.588	0.789	0.547	0.696	0.462	0.822	0.575	0.587	0.511	0.659	0.507	0.806	**0.633**	0.622	0.440	0.530	0.514
Llama2	0.583	0.570	0.512	0.544	0.500	0.500	0.463	0.513	0.568	0.416	0.607	0.500	**0.631**	0.424	0.626	0.404	0.563	**0.607**
HaluEval2-finance																		
Qwen	0.739	0.506	0.737	0.516	0.689	0.421	**0.740**	0.481	0.614	**0.545**	0.671	0.504	0.703	0.526	0.661	0.520	0.525	0.473
Llama2	0.567	**0.559**	0.502	0.553	0.500	0.500	0.568	0.538	**0.630**	0.548	0.628	0.522	0.571	0.471	0.599	0.524	0.543	0.442

Table 7: Detection results (the number of predicted positives and recall, higher is better) on all 200 erroneous samples with Qwen2 on MedHallu-en. We also report the number of false positives (lower is better) and the precision metric (higher is better). **Red**: best. Blue: second best.

Severity	SelfElicit	IO	ContextIO	HistoryIO	CoT	CoVE	FaR	SelfChkGPT	ChatProtect
overall↑	**176 (0.88)**	135 (0.68)	114 (0.57)	87 (0.43)	81 (0.41)	76 (0.38)	120 (0.60)	122 (0.61)	163 (0.81)
high-risk↑	**43 (0.86)**	24 (0.48)	25 (0.50)	16 (0.32)	20 (0.40)	13 (0.26)	25 (0.50)	24 (0.48)	38 (0.76)
medium-risk↑	**77 (0.87)**	64 (0.72)	51 (0.57)	36 (0.40)	34 (0.38)	32 (0.36)	72 (0.81)	52 (0.58)	71 (0.80)
low-risk↑	**56 (0.92)**	47 (0.77)	38 (0.62)	35 (0.57)	27 (0.44)	31 (0.51)	23 (0.38)	46 (0.75)	54 (0.89)
False Positive↓	**351**	482	474	415	359	372	547	489	584
Precision↑	**0.33**	0.22	0.19	0.17	0.18	0.17	0.18	0.20	0.22

Table 8: Detailed ablation metrics of all variants.

Dataset	LLM	w/o context		w/o elicit		w/o sample		w/o conflict		full	
		S	R	S	R	S	R	S	R	S	R
MedHallu-zh	Qwen	0.743	0.621	0.796	0.615	0.803	0.652	0.808	0.668	0.810	0.671
	ChatGLM	0.787	0.613	0.789	0.614	0.788	0.615	0.788	0.615	0.798	0.622
MedHallu-en	Qwen	0.784	0.622	0.795	0.624	0.797	0.632	0.796	0.629	0.803	0.656
	Llama2	0.712	0.572	0.728	0.569	0.727	0.564	0.719	0.572	0.748	0.582
WikiBio	Qwen	0.527	0.628	0.547	0.622	0.543	0.614	0.562	0.632	0.594	0.653
	Llama2	0.516	0.559	0.521	0.572	0.527	0.541	0.542	0.639	0.568	0.705
HaluEval2	Qwen	0.729	0.516	0.723	0.510	0.780	0.532	0.781	0.542	0.789	0.564
	Llama2	0.513	0.529	0.512	0.523	0.516	0.526	0.518	0.531	0.521	0.545

Prompt for identifying named entities and extracting knowledge statements

You are a knowledge extractor. Your task is to identify named entities from the given sentences and extract the knowledge points related to these entities.
Steps:

1. For each sentence, identify the named entities within. Named entities include, but are not limited to: {{entity types}}
   Please use the format "Named entities in sentence 1: Entity 1 (Type 1)" to list all the named entities you find.
2. For each identified named entity, extract all the related knowledge points, ensuring the semantic integrity of the points, and that they can be understood independently from the original sentence. If independent knowledge points cannot be extracted, please return the original sentence directly. Please use the format "Knowledge points in sentence 1: [Knowledge point 1][Knowledge point 2]"to list all the knowledge points you find.

{{few shot}}
Your task is to provide named entities and knowledge points based on the following sentence:
{{sentence}}
Named entities:

Figure 10: Prompt for identifying named entities and extracting knowledge statements.

G Pseudo-code

Algorithm 1 shows the pseudo-code of SelfElicit.

Table 9: Inference costs for all methods with the Qwen model. `Perform.`: average AUC metrics. `#Call`: number of LLM calls. `#Token`: number of generated tokens.

Dataset	Method	Relative Perform.↑	#Call↓	Relative #Call↓	#Token↓ (k)	Relative #Token↓
MedHallu-zh	IO	-7.9%	7,552	-39.4%	390	-61.7%
	ContextIO	-8.1%	7,552	-39.4%	399	-60.9%
	HistoryIO	-3.1%	7,552	-39.4%	370	-63.7%
	CoT	-18.4%	7,552	-41.6%	934	-8.5%
	CoVE	-22.7%	36,852	+196.0%	1,828	+79.2%
	FaR	-7.1%	14,104	+13.3%	2,309	+126.4%
	SelfCheckGPT	-32.5%	130,912	+951.3%	13,711	+1244.0%
	ChatProtect	-30.5%	138,758	+1014.3%	5,703	+459.0%
	SelfElicit	-	12,452	-	1,020	-
MedHallu-en	IO	-5.2%	7,422	-37.3%	636	-54.7%
	ContextIO	-7.0%	7,422	-37.3%	657	-53.2%
	HistoryIO	-1.2%	7,422	-37.3%	489	-65.2%
	CoT	-20.1%	7,422	-37.3%	1,296	-7.8%
	CoVE	-31.6%	38,696	+226.7%	2,484	+76.8%
	FaR	-5.9%	14,104	+19.1%	2,752	+95.9%
	SelfCheckGPT	-10.5%	131,066	+1006.5%	10,828	+670.9%
	ChatProtect	-30.8%	164,010	+1284.6%	6,398	+355.5%
	SelfElicit	-	11,845	-	1,405	-
WikiBio	IO	-7.5%	1,908	-61.9%	159	-70.4%
	ContextIO	-10.6%	1,908	-61.9%	175	-67.4%
	HistoryIO	-7.3%	1,908	-61.9%	219	-59.3%
	CoT	-14.5%	1,908	-61.9%	526	-2.1%
	CoVE	-15.5%	12,619	+152.3%	1,156	+114.9%
	FaR	-15.4%	5,724	+14.4%	1,399	+160.0%
	SelfCheckGPT	-5.7%	10,730	+114.5%	870	+61.7%
	ChatProtect	-6.5%	86,178	+1622.9%	2,410	+348.0%
	SelfElicit	-	5,002	-	538	-
HaluEval2	IO	-8.0%	1,863	-18.3%	113	-58.6%
	ContextIO	-16.2%	1,863	-18.3%	148	-46.0%
	HistoryIO	-3.4%	1,863	-18.3%	153	-44.0%
	CoT	-16.4%	1,863	-18.3%	559	+103.5%
	CoVE	-11.9%	9,641	+322.9%	1,823	+563.3%
	FaR	-2.7%	5,589	+145.1%	2,244	+716.3%
	SelfCheckGPT	-13.6%	11,400	+400.0%	1,662	+504.5%
	ChatProtect	-22.7%	60,240	+2542.1%	1,047	+280.8%
	SelfElicit	-	2,280	-	274	-

Algorithm 1: Self-elicitation Procedure.

```
Input  : Sentences $\{r_1, r_2, \cdots\}$, a language model LM, a NLI model NLI.
Output : Sentence-wise non-factual scores $\hat{y}_1, \hat{y}_2, \cdots$, and response-wise score $\hat{Y}$.
/* Extract entities and statements */
$s_1, s_2, \cdots, V_1, V_2, \cdots \leftarrow \mathrm{LM}(r_1, r_2, \cdots)$;
/* Graph-guided self-elicitation */
Initialize graph $\mathcal{G}_0$ with vertex set $\mathbb{V} \leftarrow V_1 \cup V_2 \cup \cdots$, and edge set $\mathbb{E}_0 \leftarrow \varnothing$;
for $s_i \in \{s_1, s_2, \cdots\}$ do
    /* Knowledge sampling */
    for $k \in [\alpha, \beta]$ do
        Sample $\hat{\mathbb{E}}_i(k)$ from graph $\mathcal{G}_{i-1}$ with related vertices $\hat{\mathbb{V}}_i(k)$;
    end
    Aggregate all $\hat{\mathbb{E}}_i(k)$ and linearize to context $C_i$;
    /* Fact-evaluation & Elicitation */
    Evaluate $s_i$ given context $C_i$ with LM, obtaining score $\hat{p}_i$ and reflections $o_i^{refl}$;
    /* Graph Update */
    Obtain new edges $\tilde{\mathbb{E}}_i$ from reflection $o_i^{refl}$;
    $\mathbb{E}^{orig} \leftarrow \mathbb{E}_{i-1}$;
    for $e \in \tilde{\mathbb{E}}_i$ do
        $\mathbb{E}^{temp} \leftarrow \varnothing$;
        if $e.vertices == \bar{e}.vertices, \exists \bar{e} \in \mathbb{E}^{orig}$ then
            rel $\leftarrow \mathrm{NLI}(e, \bar{e})$;
            if rel is "entail" then Add $e$ to $\mathbb{E}^{temp}$ ;
            else if rel is "neutral" then Add $e$ and $\bar{e}$ to $\mathbb{E}^{temp}$ ;
            else                                            /* mitigate conflicts */
                $\hat{e} \leftarrow \mathrm{LM}(e, \bar{e})$;
                Add $\hat{e}$ to $\mathbb{E}^{temp}$;
        else
            Add $e$ to $\mathbb{E}^{temp}$;
        end
        $\mathbb{E}^{orig} \leftarrow \mathbb{E}^{temp}$
    end
    Update graph $\mathcal{G}_i$ with edge set $\mathbb{E}^{orig}$;
end
Obtain sentence predictions $\hat{y}$ by aggregating scores from statements;
Obtain response prediction $\hat{Y}$ by aggregating scores from sentences;
```

Prompt for evaluation and reflection

Context: {{CONTEXT}}
Description: {{SENTENCE}}
Is the above description:
A True
B False
C Not sure
Choose your option from A, B and C and explain why:

Figure 11: Prompt for evaluation and reflection.

Prompt for detecting the relation between two statements

Please determine the semantic relationship between the following two sentences. There are three possible types of relationships:
1. [entail]: The content of the two sentences is the identical, describing the same aspect of the same object, with consistent content.
2. [contradict]: The two sentences describe the same aspect of the same object, but the content is directly opposite, presenting a contradiction.
3. [neutral]: The two sentences describe different objects, or different aspects of the same object, and can coexist.
Please analyze sentence A and sentence B, and choose one of the relationships. Please briefly explain your reasoning.
Sentence A: {{SENTENCE_A}}
Sentence B: {{SENTENCE_B}}
Judgment result:

Figure 12: Prompt for detecting the relation between two statements.

Prompt for mitigating the conflicts between two statements

Please read the following two sentences.
These two sentences describe the same aspect of the same object, but their content is contradictory.
Your task is to judge which sentence is more accurate based on your own understanding.
Sentence A: {{SENTENCE_A}}
Sentence B: {{SENTENCE_B}}
Judging criteria:
Please consider the logic and factual basis of the sentences. Choose the sentence you think is correct and select from the following two options:
[Sentence A is correct]
[Sentence B is correct]

Figure 13: Prompt for mitigating the conflicts between two statements.

Original Sample

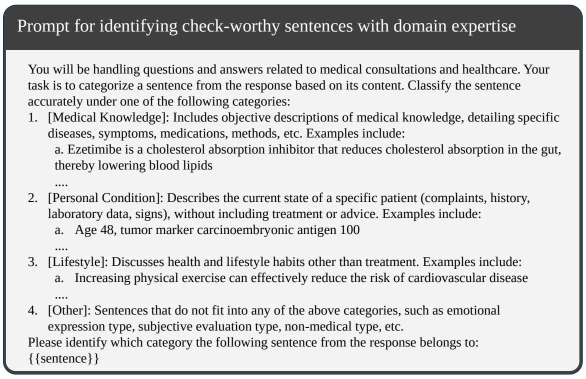
User Query:
When should Gliclazide be taken? Why?
Candidate Response:
Gliclazide is an oral hypoglycemic medication. It is suitable for adult type 2 diabetes patients whose blood sugar cannot be controlled by diet alone. Generally, Gliclazide can be taken at any time of the day, regardless of whether it is on an empty stomach or after meals. This is mainly because the mechanism of action of Gliclazide is to lower blood sugar by stimulating the secretion of insulin from pancreatic β-cells, and this process is not affected by meal intake. It should be noted that the specific timing and dosage of medication should be determined based on individual differences and the doctor's advice. In addition, during the use of Gliclazide, it is necessary to regularly monitor blood sugar levels and adjust the dosage according to medical advice. Do not increase or decrease the dosage or stop taking the medication on your own to avoid adverse reactions such as hypoglycemia.
Human Label: × Incorrect. Gliclazide should be taken before each meal.

Preprocessed Sample

Sentences:
1. Gliclazide is an oral hypoglycemic medication✓
2. It is suitable for adult type 2 diabetes patients whose blood sugar cannot be controlled by diet alone✓
3. Generally, Gliclazide can be taken at any time of the day, regardless of whether it is on an empty stomach or after meals×
4. This is mainly because the mechanism of action of Gliclazide is to lower blood sugar by stimulating the secretion of insulin from pancreatic β-cells, and this process is not affected by meal intake×
5. It should be noted that the specific timing and dosage of medication should be determined based on individual differences and the doctor's advice✓
6. In addition, during the use of Gliclazide, it is necessary to regularly monitor blood sugar levels and adjust the dosage according to medical advice✓
7. Do not increase or decrease the dosage or stop taking the medication on your own to avoid adverse reactions such as hypoglycemia✓

Claims:
1. Gliclazide is an oral hypoglycemic medication✓
2. Gliclazide is suitable for adult type 2 diabetes patients whose blood sugar cannot be adequately controlled by diet alone✓
3. Gliclazide can be taken at any time of the day×
4. Gliclazide can be taken either on an empty stomach or after meals×
5. The mechanism of action of Gliclazide is to lower blood glucose by stimulating pancreatic β-cells to secrete insulin✓
6. The action process of Glargine is not affected by food intake✓
7. The specific timing and dosage of Gliclazide medication should be determined based on individual differences and the doctor's recommendations✓
8. During the use of Gliclazide, it is necessary to regularly monitor blood sugar levels and adjust the dosage according to the doctor's instructions✓
9. Do not adjust the dosage or discontinue the medication on your own when using Gliclazide✓
10. Adjusting the dosage of Gliclazide on your own may lead to adverse reactions such as hypoglycemia✓

Table 10: A sample from MedHallu-en. ✓refers to factual and ×refers to non-factual.



Prompt for identifying check-worthy sentences with domain expertise

You will be handling questions and answers related to medical consultations and healthcare. Your task is to categorize a sentence from the response based on its content. Classify the sentence accurately under one of the following categories:
1. [Medical Knowledge]: Includes objective descriptions of medical knowledge, detailing specific diseases, symptoms, medications, methods, etc. Examples include:
   a. Ezetimibe is a cholesterol absorption inhibitor that reduces cholesterol absorption in the gut, thereby lowering blood lipids
   ....
2. [Personal Condition]: Describes the current state of a specific patient (complaints, history, laboratory data, signs), without including treatment or advice. Examples include:
   a. Age 48, tumor marker carcinoembryonic antigen 100
   ....
3. [Lifestyle]: Discusses health and lifestyle habits other than treatment. Examples include:
   a. Increasing physical exercise can effectively reduce the risk of cardiovascular disease
   ....
4. [Other]: Sentences that do not fit into any of the above categories, such as emotional expression type, subjective evaluation type, non-medical type, etc.

Please identify which category the following sentence from the response belongs to:
{{sentence}}

Figure 14: Prompt for identifying check-worthy sentences with domain expertise for MedHallu-zh and MedHallu-en.