JointCQ: Improving Factual Hallucination Detection with Joint Claim and Query Generation

Fan Xu[1], Huixuan Zhang[1], Zhenliang Zhang[1], Jiahao Wang[2], Xiaojun Wan[1]
[1]Wangxuan Institute of Computer Technology, Peking University
[2]Trustworthy Technology and Engineering Laboratory, Huawei
{xufan2000,wanxiaojun}@pku.edu.cn, {zhanghuixuan,zhenliang}@stu.pku.edu.cn, wangjiahao50@huawei.com

Abstract

Current large language models (LLMs) often suffer from hallucination issues, i,e, generating content that appears factual but is actually unreliable. A typical hallucination detection pipeline involves response decomposition (i.e., claim extraction), query generation, evidence collection (i.e., search or retrieval), and claim verification. However, existing methods exhibit limitations in the first two stages, such as context loss during claim extraction and low specificity in query generation, resulting in degraded performance across the hallucination detection pipeline. In this work, we introduce JointCQ [1], a joint claim-and-query generation framework designed to construct an effective and efficient claim-query generator. Our framework leverages elaborately designed evaluation criteria to filter synthesized training data, and finetunes a language model for joint claim extraction and query generation, providing reliable and informative inputs for downstream search and verification. Experimental results demonstrate that our method outperforms previous methods on multiple open-domain QA hallucination detection benchmarks, advancing the goal of more trustworthy and transparent language model systems.

1 Introduction

Large Language Models (LLMs) have achieved remarkable success across a wide range of natural language generation (NLG) tasks, including open-domain question answering (QA) (Kamalloo et al., 2023). However, despite their impressive capabilities, LLMs are susceptible to factual hallucinations, where models generate responses that appear plausible but are factually incorrect, as mentioned in multiple previous works (Huang et al., 2023; Ji et al., 2023; Zhang et al., 2023b). This issue poses significant challenges for users who rely on LLMs for accurate information, raising critical concerns about the reliability and accountability of AI-generated content. As LLMs continue to advance and become increasingly integrated into real-world applications, addressing hallucinations is crucial to ensuring their trustworthiness and practical utility (Pal et al., 2023; Dahl et al., 2024). Detecting factual hallucinations in generated content has thus become a critical area of research.

Prior studies have explored various detection methods with distinct limitations. Some approaches rely on self-verification techniques, such as prompting LLMs or sampling generations (Manakul et al., 2023; Ni et al., 2024), which may inherit the same biases or knowledge gaps as the original model. Others analyze internal model states or generation probabilities (Zhang et al., 2023a; Azaria and Mitchell, 2023), but these signals can be opaque and model-specific. In contrast, retrieval-based methods, which systematically search for relevant external information and compare it with generated content, have proven particularly effective, as they provide concrete, verifiable evidence for hallucination detection (Cheng et al., 2024; Chern et al., 2023). In fields where reliable information is essential, such as healthcare, finance, scientific research, or any scenario involving internal or sensitive data, retrieval-based methods become particularly essential. Existing retrieval-based detection methods for open-domain question answering typically decompose responses, generate queries and perform evidence retrieval and claim verification. However, these approaches frequently struggle with suboptimal decomposition (Metropolitansky and Larson, 2025; Wanner et al., 2024; Ullrich et al., 2025) and query generation (Jeong et al., 2024), limiting their effectiveness.

To effectively detect factual hallucinations in language model outputs, it is essential to first generate grounded claims along with their corresponding retrieval-oriented queries. This relies a model

[1]https://github.com/pku0xff/JointCQ

Findings of the Association for Computational Linguistics: ACL 2026, pages 1138–1159

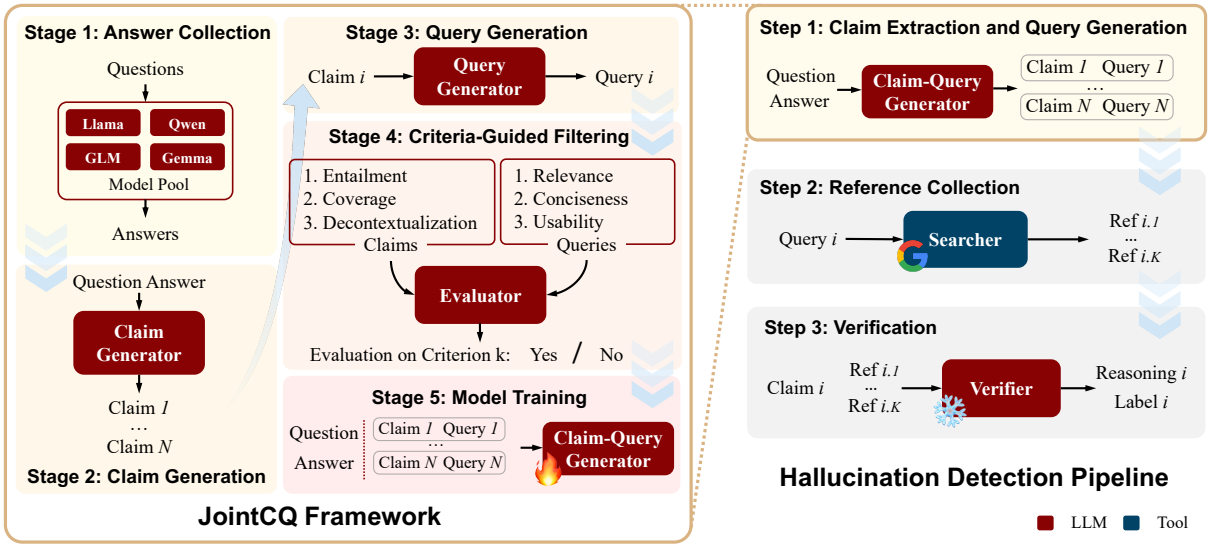

Figure 1: Overview of the JointCQ framework (left) and hallucination detection pipeline (right). The claim-query generator is built with the JointCQ framework and can jointly generate claims and their corresponding queries in a single inference step.

trained on high-quality and well-aligned claim-query pairs. Therefore, we propose **JointCQ**, a comprehensive framework that includes both the construction of training data and the training of a joint claim-query generation model. The framework first uses an LLM to generate candidate claims and queries, then applies a rigorous filtering process to ensure data quality. The resulting filtered data is used to finetune a language model that can produce reliable claims and the corresponding queries in a single inference step.

The core strength of JointCQ lies in its criteria-guided data filtering process. Rather than relying on loosely aligned or noisy data, we apply a dual evaluation procedure that filters claims and queries independently. For claims, we assess entailment, coverage, and decontextualization. For queries, we evaluate relevance, conciseness, and usability to ensure that they support effective retrieval and align closely with the associated claims. As a result, the JointCQ framework ensures high-quality training data and enables a more effective joint claim-query generator. This generator serves as a solid foundation for downstream hallucination detection process. Additionally, our framework is fully built upon open-source models and supports both English and Chinese. Experiments on open-domain QA hallucination detection benchmarks demonstrate that our method outperforms strong baselines on both languages, advancing the development of more trustworthy and transparent language model systems.

To summarize, our main contributions are:

1. We propose **JointCQ**, a framework that can train a model capable of generating both factual claims and their corresponding search queries in a single inference for factual hallucination detection. The framework is fully built on open-source models, ensuring low cost, high accessibility, and ease of deployment.
2. We design a dual-stage, criteria-guided filtering strategy to construct high-quality training data in JointCQ, ensuring the model is trained on accurate and well-aligned claim-query pairs.
3. Experimental results on multiple open-domain QA hallucination detection benchmarks demonstrate that JointCQ substantially improves the factual hallucination detection performance, surpassing several strong baselines.

2 Hallucination Detection Task

2.1 Task Formulation

Given a question and a corresponding answer generated by a language model, our goal is to detect factual hallucinations at the claim level. We adopt the definition of a factual claim from Ni et al. (2024), where a claim is a statement explicitly presenting verifiable facts. Here, a fact is an assertion that can be objectively verified as true or false based on empirical evidence or reality. This claim-level formulation allows for fine-grained hallucination detection. It also supports more targeted verifica-

tion and modular processing.

Formally, the task can be described as:

- **Input**: A natural language question q and a model-generated answer a that may contain correct information, hallucinations, or unverifiable content.
- **Output**: A set of factual claims $\{c_1, c_2, \ldots, c_N\}$ extracted from (q, a), where each claim c_i is assigned with a factuality label $l_i \in \{Correct, Hallucinated, Unverifiable\}$ indicating its status based on external evidence.

2.2 Pipeline Components

A standard hallucination detection pipeline typically consists of four sequential steps (Min et al., 2023; Chern et al., 2023; Fatahi Bayat et al., 2023; Wei et al., 2024; Cheng et al., 2024): (1) response decomposition, (2) query generation[2], (3) evidence retrieval, and (4) factual verification. However, this pipeline design often leads to issues such as missing factual details, loss of context, and insufficiently targeted queries.

To address these issues, we redesign the pipeline by unifying the first two stages into a single step using our proposed **JointCQ** framework. As shown in the right part of Figure 1, given a question and its answer, the claim-query generator jointly extracts factual claims and generates corresponding queries. In our implementation, the searcher sends these queries to Google Search via the Serper API[3] and retrieves the top-10 snippets as evidence. Since these queries are formulated as natural language (see Section 3.2.3), the retrieval component is highly modular, making it feasible to replace Google Search with other search engines or local corpora. Finally, a verifier implemented with `Qwen3-14B`[4], assesses each claim's factuality against the retrieved snippets. Appendix B provides additional information on the implementation of hallucination detection pipeline.

3 JointCQ Framework

3.1 Overview

This section presents the JointCQ framework, designed to enhance hallucination detection by optimizing the claim extraction and query generation stage (Figure 1). Central to our approach is the construction of high-quality, well-aligned claim-query training data through a rigorous, criteria-guided filtering process, ensuring effective and efficient supervision. The filtered data is then used to train a joint claim-query generation model capable of producing claim-query pairs in a single inference step.

3.2 Data Synthesis

3.2.1 Data Sourcing

The `question` segment of the ANAH-v2 dataset (Gu et al., 2024) serves as the core data source. This dataset consists of questions and reference documents, but does not include hallucination labels. We leverage a diverse set of mainstream large language models to generate corresponding answers: `Qwen2.5-7B-Instruct` (Qwen et al., 2025), `Llama-3.1-8B-Instruct` (Grattafiori et al., 2024), `gemma-3-4b-it` (Team et al., 2025) and `glm-4-9b-chat` (GLM et al., 2024). This ensures the richness of answer variations, thereby laying a comprehensive foundation for extracting diverse factual claims. Consequently, this stage yields a collection of question–answer pairs that serve as input for subsequent stages of supervised data construction.

3.2.2 Claim Synthesis

Claim extraction is performed using a 3-shot prompting strategy to guide the claim generation model, `Qwen3-32B` (Yang et al., 2025). In-context examples are constructed from the same dataset described in the previous section, with output segments manually written. For each QA pair, we first retrieve the top-3 examples with the highest semantic similarity (measured by the `paraphrase-multilingual-mpnet-base-v2` embedding model (Reimers and Gurevych, 2019)) and the top-3 examples with the most similar answer length. From this candidate pool of up to six examples, we randomly sample three as the final in-context examples.

The model is instructed to generate clear, factual, and self-contained claims, excluding subjective or ambiguous content. By applying this prompting process, we extract a set of factual claims $\{c_1, \ldots, c_N\}$ from each QA pair.

3.2.3 Query Synthesis

Query generation adopts a 3-shot prompting strategy, selecting three random examples. The query

[2]Some approaches simplify this step by extracting keywords or directly reusing decomposed segments as queries.

[3]https://serper.dev

[4]Other LLMs, especially larger models, will work well or even better in this step, but for cost and efficiency consideration, we simply use Qwen3-14B here.

generator is implemented with `Qwen3-32B` as well. For each claim c_i, a search query q_i is generated, bridging the gap between extracted claims and the evidence retrieval stage. For more details on the data synthesis implementation, please refer to Appendix A.1.

3.3 Criteria-Guided Filtering

To improve the quality of claims and queries in our training dataset, we use a filtering process on both elements. The process guarantees that each claim is grounded in input QA pairs and clearly stated, while each query is effective for finding relevant information. Examples of passed and failed claims and queries on each criterion are shown in Table 10 in Appendix A.2.

3.3.1 Claim Evaluation Criteria

For the selection of claims, we adopt and modify the criteria mentioned by Metropolitansky and Larson (2025):

- ***Entailment***: *The content of the claims should be fully supported by the source text, i,e, the question and answer.*
  Unlike settings where claims are derived solely from answers, we treat the question as an essential part of the context. This is because many answers are underspecified on their own, and only make complete sense when interpreted alongside the question.
- ***Coverage***: *The extracted claims should capture all the verifiable factual information in the source text.*
  This helps avoid selective reporting or omission of fact-related information.
- ***Decontextualization***: *The claim should be understandable on its own, without requiring additional context.*
  This criterion follows principles from sentence decontextualization research (Choi et al., 2021), which emphasize the portability and semantic completeness of isolated textual statements.

While grounded in similar theoretical foundations, our use case and filtering process differs from the evaluation framework of Metropolitansky and Larson (2025), where claims are directly used as search queries to retrieve supporting evidence. We introduce an additional step by generating a separate query for each claim. This query is optimized for external information retrieval (e.g., from a search engine) and is evaluated using its own set of criteria. This distinction is important: it allows us to maintain the factual clarity and independence of each claim while tailoring the retrieval process through purpose-built, query-specific formulations. By separating claim construction from query design, we are able to better control for both the verifiability of the content and the effectiveness of the retrieval process. This separation lead to a total different definition of decontextualization.

3.3.2 Query Evaluation Criteria

Unlike claims, query evaluation emphasizes retrieval effectiveness and search-oriented design. Our formulation of query criteria draws from information retrieval theory (Schütze et al., 2008; Cronen-Townsend et al., 2002). The criteria are as follows:

- ***Relevance***: *The query directly relates to the claim, addressing its content, implications, or underlying assumptions.*
  This criterion ensures that retrieved information is semantically aligned with the claim, thereby reducing the inclusion of off-topic or tangential evidence. It serves as a basic but essential filter for maintaining consistency between the claim and external knowledge sources.
- ***Conciseness***: *The query should be clear and focused on the core information. Avoid multiple complex ideas or detailed descriptions in one query.*
  This criterion corresponds to the query clarity principle in IR literature, where shorter and clearer queries can yield more relevant results.
- ***Usability***: *The query should use natural, fluent, and easily readable language that can yield relevant and accurate results from Google Search.*
  This criterion captures the practical need for queries to be interpretable by real-world search engines. Natural-sounding queries are more likely to elicit high-quality results, both in human-centered and automated search scenarios.

3.3.3 Evaluation Protocol Design

To implement the filtering at scale, we design a hybrid evaluation protocol that leverages the capabilities of the `Qwen3-32B` language model. We separate the evaluation procedures for different criteria to minimize cross-dimensional interference and maximize reliability.

For entailment and coverage, we conduct evaluation in a batch-oriented manner, where each batch

corresponds to the full set of claims extracted from a single QA pair. This provides the model with sufficient context.

By contrast, decontextualization is evaluated at the individual claim level, with each claim presented to the model in isolation, absent accompanying claims. This setup directly tests whether the claim remains semantically self-sufficient.

Similarly, evaluation of queries is conducted on an individual basis, with each query-claim pair assessed separately. This ensures a localized evaluation of query quality, unimpeded by interactions with other queries or external context. Appendix A.2 offers a more thorough description of the criteria-guided filtering implementation.

3.3.4 Manual Validation

To more accurately assess the reliability of the automatic filtering and ensure the quality of the training data, we conduct targeted human evaluation of the filtering pipeline. Following the structure of our model-based filtering, we divide the annotation into three subtasks:

1. Claim set evaluation: Given a question, an answer, and its extracted claims, evaluate the entailment and coverage of the claim set.
2. Decontextualization quality: Given a claim, assess whether it is properly decontextualized.
3. Query quality: Given a claim and its query, evaluate the query's relevance, conciseness, and usability.

For each subtask, we independently sample 100 unfiltered examples and recruit one qualified annotator per subtask. In Table 1, we report both the consistency between human judgments and model predictions, as well as the model's false positive rate when treating human annotations as the gold standard. Overall, the model performs effective filtering, though it is less strict on coverage and tends to miss finer-grained coverage issues. Appendix D.1 presents the annotation process in detail.

Criterion	Consistency (%)	FP (%)
Entailment	85	6
Coverage	69	27
Decontextualization	88	7
Relevance	98	1
Conciseness	85	14
Usability	95	5

Table 1: Manual validation results of the filtering criteria. The metrics are human-model consistency rate and model false positive (FP) rate.

3.4 Model Training

3.4.1 Data Preparation

To mitigate bias toward a specific claim count of each QA pair, we stratify samples by their claim count and enforce per-group sampling limits. After stratified sampling, random selection fills remaining quotas, producing a final dataset of 1,000 samples for each language with moderately balanced claim count distributions. We partition each language subset into training and test sets (9:1 ratio), resulting in 1,800 training and 200 validation samples.

3.4.2 Training Details

We fine-tune the `Qwen2.5-14B-` Instruct (Qwen et al., 2025) model as our Claim-Query Generator, leveraging its strong instruction-following aptitude and computational efficiency for this task. Training runs for 1 epoch on synthetic (claim, query) pairs with a batch size of 128, optimized for memory efficiency on 4×NVIDIA H100 GPUs (80GB VRAM) using DeepSpeed Zero-3 for distributed training. Hyperparameters include a 1e-5 learning rate (10% linear warmup), and bfloat16 mixed-precision training with gradient checkpointing.

4 Experiment Setup

4.1 Test Sets

We evaluate our method on two publicly available benchmark datasets across different domains and languages:

- **ANAH** (Ji et al., 2024)[5]: A bilingual dataset with sentence-level hallucination annotations from LLM responses. We sample 500 QA pairs per language for a 1,000-sample test set supporting both response- and sentence-level evaluation. This size is relatively large compared to similar prior works (Chern et al., 2023; Cheng et al., 2024), allowing for reliable assessment.
- **HalluQA** (Cheng et al., 2023): A Chinese hallucination detection benchmark for QA task with binary, response-level labels. We use all the 206 fact-related samples for our experiments, following the setup in HaluAgent (Cheng et al., 2024).

[5]ANAH is a totally different dataset from the ANAH-v2 mentioned in Section 3.2.1.

These test sets cover both English and Chinese, and support multi-granularity hallucination analysis, providing a comprehensive benchmark for evaluating the generalization and robustness of hallucination detection methods.

4.2 Baselines

We compare our framework with several strong base LLMs and hallucination detection methods.

- **Advanced LLMs**: Gemini-2.5-Pro (Comanici et al., 2025), GPT-5 (Singh et al., 2025), GPT-4.1 (OpenAI et al., 2024), Deepseek-V3.1-Terminus (DeepSeek-AI, 2024), and DeepSeek R1 (DeepSeek-AI et al., 2025). These state-of-the-art general-purpose LLMs inherently possess the ability to perform hallucination detection.
- **SelfCheckGPT** (Manakul et al., 2023): A classical hallucination detection method that detects hallucinations by generating multiple responses from a language model and checking for consistency across them.
- **FacTool** (Chern et al., 2023): A tool-augmented framework designed for factual error detection across diverse generative tasks.
- **HaluAgent** (Cheng et al., 2024): An autonomous hallucination detection framework built on small open-source models, integrating multiple tools for fact-checking.

4.3 Evaluation Metrics

We use **Accuracy** and hallucination **F1 score** for both sentence- and response-level evaluation. Unverifiable or failed samples are treated as no hallucination, similar to the setup in FacTool (Chern et al., 2023). Evaluation results for only the verifiable samples are in Appendix C.

For sentence-level evaluation, claim c_j is aligned to response sentence s_i when: (1) s_i is most semantically similar to c_j, and (2) cosine similarity[6] exceeds threshold $\theta = 0.5$ [7].

Let R denote the set of sentences in a response. The aligned claims for s_i are defined as:

$$C(s_i) = \{c_j \mid s_i = \arg\max_{s_k \in R} \text{sim}(s_k, c_j) \wedge \text{sim}(s_i, c_j) \geq \theta\}.$$

Hallucination labels are aggregated hierarchically:

$$H(s_i) = \mathbb{I}[\exists c_j \in C(s_i) : h(c_j) = 1],$$
$$H(r) = \mathbb{I}[\exists s_i \in R : H(s_i) = 1],$$

where $\mathbb{I}[\cdot]$ is the indicator function. This ensures consistent evaluation across annotation granularities. Further details about the experiment setup and results can be found in Appendix C.

5 Results and Analysis

5.1 Main Results

Table 2 presents the response-level evaluation results. Our method achieves competitive results, with the highest scores on ANAH-overall and sub-optimal scores on HalluQA. Meanwhile, it results in the fewest unverifiable samples and exhibits better usability. FacTool shows lower accuracy on HalluQA but performs moderately on ANAH. While HaluAgent-13B achieves high accuracy on ANAH-en and HalluQA, its performance drops significantly on ANAH-zh, suggesting language- and domain-dependent limitations. Notably, general large language models (e.g., Gemini-2.5-Pro) attain optimal results on several individual metrics. However, these models lag behind specialized hallucination detection approaches in interpretability and fail to provide fine-grained diagnostic insights.

Table 3 presents sentence-level hallucination detection results on the ANAH dataset. Our method achieves state-of-the-art performance across all settings, attaining the highest scores in both English (ANAH-en: 80.14% Acc/70.99% F1) and Chinese (ANAH-zh: 76.16% Acc/71.10% F1) verifiable samples, with consistent advantages of +5~8% accuracy and +3~4 F1 points over FacTool.

Overall, the experimental results demonstrate that our proposed framework outperforms the baseline methods in most cases, whether evaluating at the response level or the sentence level. Our framework shows better accuracy and F1 scores, indicating its strong capability in detecting factual hallucinations on the open-domain QA task.

5.2 Necessity of Queries

Previous work (Metropolitansky and Larson, 2025) state that claims are used to retrieve relevant information from sources, which is different from our settings of using additional queries. To assess the importance of the query generation step, we conduct an ablation study where the generated

[6] Texts are embedded with `paraphrase-multilingual-mpnet-base-v2` (Reimers and Gurevych, 2019).

[7] The threshold is empirically chosen to filter out pairs with low semantic relatedness, as text pairs with cosine similarity below 0.5 are typically considered non-matching in semantic similarity tasks.

	ANAH-en			ANAH-zh			ANAH-overall			HalluQA		
	Acc	F1	N unv.	Acc	F1	N unv.	Acc	F1	N unv.	Acc	F1	N unv.
Gemini-2.5-Pro	**76.40**	74.12	-	68.20	70.72	-	72.30	72.27	-	**81.07**	82.51	-
GPT-5	74.60	73.92	-	66.00	69.64	-	70.30	71.63	-	80.58	82.14	-
GPT-4.1	71.80	65.01	-	61.40	56.43	-	66.60	60.52	-	72.82	70.53	-
Deepseek-V3.1	51.60	47.84	-	58.40	58.06	-	55.00	53.12	-	67.96	71.05	-
DeepSeek R1	61.40	42.73	-	61.40	58.13	-	61.40	51.63	-	76.70	74.19	-
SelfCheckGPT	70.20	74.35	-	67.60	75.89	-	69.80	75.18	-	56.31	68.97	-
FacTool	74.20	**77.33**	13	68.60	76.46	11	71.40	76.86	24	56.80	46.71	12
HaluAgent-13B	72.80	70.82	21	67.20	67.97	29	70.00	69.30	50	78.16*	**83.75***	-
Ours	75.80	76.95	5	**72.60**	**77.58**	11	**74.20**	**77.29**	16	80.58	83.05	5

Table 2: Response-level evaluation results. Acc and F1 values are reported in percentage. The results of HaluAgent-13B on HalluQA dataset comes from the paper (Cheng et al., 2024). "N unv." denotes the number of unverifiable samples. The full name of Deepseek-V3.1 is Deepseek-V3.1-Terminus. Bold text denotes the optimal result, and underlined text denotes the suboptimal result.

	ANAH-en		ANAH-zh		ANAH-all	
	Acc	F1	Acc	F1	Acc	F1
FacTool	74.64	67.57	68.18	68.02	71.75	67.80
SelfCheckGPT	74.32	69.57	67.34	67.84	71.24	68.72
Ours	**80.14**	**70.99**	**76.16**	**71.10**	**78.36**	**71.04**
w/o filtering	77.63	67.32	74.85	69.55	76.39	68.42
filter c only	78.59	68.91	73.54	67.35	76.33	68.15
filter q only	78.88	68.85	75.09	68.95	77.19	68.90
w/o q	77.63	66.38	73.18	66.06	75.64	66.22
replace CQG	75.89	65.28	73.42	67.73	74.79	66.48
w/o joint	78.01	67.34	74.37	69.42	76.39	68.38

Table 3: Sentence-level evaluation of hallucination detection on ANAH dataset.

queries are replaced with claims, while keeping all other components unchanged. The experimental results are presented in Table 3, indicated as "w/o q". Compared to the complete implementation, performance drops noticeably for both Chinese and English, with a decline of 4.82 points in overall hallucination F1 score. These results underscore the necessity of incorporating a dedicated query generation step. Notably, our framework integrates claim extraction and query generation within a single inference pass, introducing minimal additional computational cost.

5.3 Effectiveness of Criteria-guided Filtering

To evaluate the impact of criteria-guided filtering, we compare three experimental settings: (1) no filtering applied to either claims or queries (w/o filtering), (2) filtering applied only to claims (filter c only), and (3) filtering applied only to queries (filter q only). The training data size and sampling strategies remain consistent with the main experiment. As shown in Table 3, omitting filtering in any configuration results in a performance decline, though the magnitude varies. This demonstrates that our curated filtering criteria enhance the quality of both claims and queries, leading to improved hallucination detection performance.

5.4 Effectiveness of Claim-Query Generator

We first conduct an ablation study by replacing the Claim-Query Generator with the separate claim synthesis and query synthesis steps with base LLMs described in Section 3.2, while keeping the rest of pipeline the same. The results, shown in Table 3 under the setting "replace CQG", indicate a clear drop in performance compared to the full JointCQ framework. Notably, even when compared to earlier ablations on criteria-guided filtering, the base synthesis approach performs worse. These findings highlight the advantage of jointly generating claims and queries in a single model inference, and further demonstrate the effectiveness of the JointCQ framework.

To more directly and rigorously isolate the effect of joint claim–query generation, we perform a second complementary ablation study. Using the same training data and the same base model as in the main experiment, we trained two separate models: one for claim extraction and one for query generation. These two models replace the joint Claim–Query Generator in the pipeline, while all other settings in the hallucination detection workflow remain identical. As shown in the row "w/o joint" in Table 3, using two task-specific models yields worse performance than the joint generation approach, demonstrating that the joint design provides benefits in both efficiency and effectiveness.

5.5 Reliability of Verifier

To evaluate the reliability of the verifier, we randomly sample 50 claims per language, along with their corresponding search results. Each claim is manually annotated as Correct, Hallucinated, or Unverifiable based on the retrieved evidence. More details about manual annotation are presented in Appendix D.2. Among the 93 claims labeled as verifiable, the model verifier `Qwen3-14B` achieves a consistency rate of 91.40% with human annotations. This result indicates that current large language models perform well on the verification task. The bottleneck in hallucination detection performance, therefore, lies in earlier stages, supporting our initial motivation. By focusing on generating higher-quality claims and queries, the proposed JointCQ framework contributes to improved detection accuracy.

5.6 Efficiency Analysis

Method	Search / jud.	Inference / sample
FacTool	2	13.40
HaluAgent	1.38	5.24
Ours	1	4.93

Table 4: Average search call per judgement and inference call per QA sample. Here judgement refers to a decision of whether the given text segment contains hallucination.

We evaluate the efficiency of the hallucination detection pipeline on 200 QA examples from the ANAH dataset. The end-to-end processing takes 599 seconds on a server with 4 NVIDIA H100 GPUs using the `vllm` engine. The main bottleneck is the reference search stage (303s), while inference remains efficient.

As shown in Table 4, our framework requires only 1 search API call per judgement and 4.93 model inferences per sample, significantly fewer than **FacTool** and comparable to **HaluAgent**. Unlike HaluAgent, which produces coarse response-level labels, JointCQ performs fine-grained, claim-level hallucination detection. In addition, while both FacTool and HaluAgent rely on APIs of closed-source models, our framework is built entirely on open-source models, offering greater accessibility and lower deployment cost.

5.7 Case Study

To illustrate the effectiveness of our framework, we present an example in Figure 2. This case illustrates two key observations. First, claims are typically more fine-grained than full sentences. Instead of assessing the entire sentence, breaking it into individual claims enables more precise identification of hallucinated content. Second, the queries are closely aligned with the specific elements of each claim, targeting the parts most likely to be incorrect. Here, the queries focus on the year of completion and the period of construction. This targeted querying improves retrieval relevance.

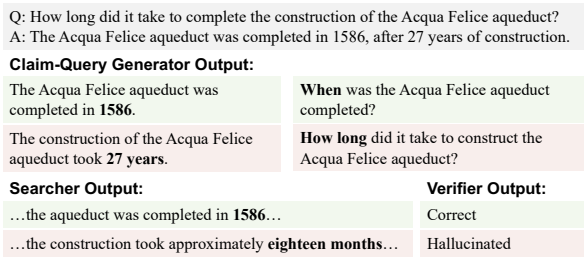

Figure 2: An example of the detection process.

6 Related Work

6.1 Factual Hallucination Detection with Web Search or Retrieval

A prominent line of research enhances factuality detection using external knowledge sources in a "retrieve-and-verify" paradigm, often decomposing content into factual units for fine-grained analysis. Min et al. (2023) propose FActScore, which verifies atomic facts against Wikipedia, offering interpretability but limited by a single-source knowledge base and explicit entity requirements. Chern et al. (2023) introduce FacTool, a unified framework across tasks such as QA, code generation, and math, while FLEEK (Fatahi Bayat et al., 2023) incorporates both detection and correction. Qin et al. (2025) propose a retrieval-augmented framework that proactively verifies false premises in queries before generation, related to our claim–query paradigm but focused on pre-generation validation. Agent-based approaches with more flexibility include SAFE (Wei et al., 2024) and HaluAgent (Cheng et al., 2024), and KnowHalu (Zhang et al., 2024) introduces a two-phase, multi-form knowledge framework with step-wise reasoning for structured factual verification.

The most closely related to our work are FacTool (Chern et al., 2023) and HaluAgent (Cheng et al., 2024). While FacTool provides a general framework across tasks, it incurs high computational cost as shown in Section 5.6. HaluAgent

adopts a more flexible agent-based approach, but it operates primarily at the response level and lacks fine-grained control over hallucination localization. In contrast, our method enables efficient, fine-grained hallucination detection.

6.2 Claim Extraction and Claim-Level Fact Checking

Claim extraction enables fine-grained factuality assessment by isolating verifiable statements. FEVERFact (Ullrich et al., 2025) provides a benchmark evaluating atomicity, fluency, and faithfulness. Metropolitansky and Larson (2025) introduces Claimify, an LLM-based method that extracts claims only when confident in interpretation. The paper also proposes a standardized framework to assess extraction quality in terms of coverage and decontextualization. We designed the training data filtering step based on the criteria introduced in this work. AFaCTA (Ni et al., 2024) leverages LLMs for consistent claim annotation, producing the PoliClaim dataset. HalluMeasure (Akbar et al., 2024) decomposes LLM outputs into atomic claims and detects hallucinations via Chain-of-Thought reasoning. However, its applicability is limited to summarization tasks and it lacks a retrieval component suited for addressing factual hallucinations. FactSelfCheck (Sawczyn et al., 2025) uses a black-box, sampling-based fact-level approach with knowledge-graph triples to enable precise claim-level detection and correction without external resources, complementing retrieval- and reasoning-based methods.

6.3 Efficient Hallucination Detection Methods

Another type of approaches aims to detect hallucinations without relying on external knowledge, prioritizing efficiency. SelfCheckGPT (Manakul et al., 2023) proposes a zero-resource, black-box method that assesses hallucination by measuring the consistency between multiple sampled outputs using metrics such as BERTScore, NLI inference, and QA agreement. To address the overconfidence or underconfidence of model-internal probabilities, Zhang et al. (2023a) introduce an uncertainty-based method using a proxy model to adjust token-level probabilities based on contextual informativeness and reliability. HaloCheck (Elaraby et al., 2023) evaluates hallucination in weaker open-source LLMs through consistency judgments among multiple responses using an NLI model. While these approaches incur low computational cost and avoid reliance on external resources, their reliability for factual verification remains limited, as they depend on internal uncertainty signals rather than grounded world knowledge.

7 Conclusion

In this work, we designed a three-stage pipeline (claim-query generation, evidence retrieval, and verification) for factual hallucination detection and introduced JointCQ, a framework that produces high-quality claims and queries to build a reliable claim-query generator. Unlike prior methods that depend on closed-source APIs, our framework is fully based on open-source models and supports both English and Chinese, making it easily accessible and broadly applicable. Experimental results demonstrate that JointCQ achieves strongest performance over multiple benchmarks, marking a step forward in building more trustworthy and transparent language model systems.

Limitations

Despite the promising results of our framework, several limitations should be noted. First, the pipeline is primarily designed for general open-domain QA tasks. While QA represents a fundamental and broadly applicable task format, extending the framework to other NLP tasks would require additional adaptation and validation. Second, our evidence retrieval component relies on Google Search, which exposes the system to the inherent limitations of the search engine. Nevertheless, leveraging such external services remains one of the most effective approaches for obtaining up-to-date and reliable information, and this strategy is commonly adopted in contemporary hallucination detection studies.

Acknowledgments

This work was supported by Beijing Natural Science Foundation (L253001), Key Laboratory of Science, Technology and Standard in Press Industry (Key Laboratory of Intelligent Press Media Technology) and National Engineering Research Center of New Electronic Publishing Technologies. We appreciate the anonymous reviewers for their helpful comments. Xiaojun Wan is the contact author.

References

Shayan Ali Akbar, Md Mosharaf Hossain, Tess Wood, Si-Chi Chin, Erica M Salinas, Victor Alvarez, and Erwin Cornejo. 2024. HalluMeasure: Fine-grained hallucination measurement using chain-of-thought reasoning. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 15020–15037, Miami, Florida, USA. Association for Computational Linguistics.

Amos Azaria and Tom Mitchell. 2023. The internal state of an LLM knows when it's lying. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 967–976, Singapore. Association for Computational Linguistics.

Qinyuan Cheng, Tianxiang Sun, Wenwei Zhang, Siyin Wang, Xiangyang Liu, Mozhi Zhang, Junliang He, Mianqiu Huang, Zhangyue Yin, Kai Chen, and 1 others. 2023. Evaluating hallucinations in chinese large language models. *arXiv preprint arXiv:2310.03368*.

Xiaoxue Cheng, Junyi Li, Xin Zhao, Hongzhi Zhang, Fuzheng Zhang, Di Zhang, Kun Gai, and Ji-Rong Wen. 2024. Small agent can also rock! empowering small language models as hallucination detector. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 14600–14615, Miami, Florida, USA. Association for Computational Linguistics.

I Chern, Steffi Chern, Shiqi Chen, Weizhe Yuan, Kehua Feng, Chunting Zhou, Junxian He, Graham Neubig, Pengfei Liu, and 1 others. 2023. Factool: Factuality detection in generative ai–a tool augmented framework for multi-task and multi-domain scenarios. *arXiv preprint arXiv:2307.13528*.

Eunsol Choi, Jennimaria Palomaki, Matthew Lamm, Tom Kwiatkowski, Dipanjan Das, and Michael Collins. 2021. Decontextualization: Making sentences stand-alone. *Transactions of the Association for Computational Linguistics*, 9:447–461.

Gheorghe Comanici, Eric Bieber, Mike Schaekermann, Ice Pasupat, Noveen Sachdeva, Inderjit Dhillon, Marcel Blistein, Ori Ram, Dan Zhang, Evan Rosen, Luke Marris, Sam Petulla, Colin Gaffney, Asaf Aharoni, Nathan Lintz, Tiago Cardal Pais, Henrik Jacobsson, Idan Szpektor, Nan-Jiang Jiang, and 3416 others. 2025. Gemini 2.5: Pushing the frontier with advanced reasoning, multimodality, long context, and next generation agentic capabilities. *Preprint*, arXiv:2507.06261.

Steve Cronen-Townsend, Yun Zhou, and W Bruce Croft. 2002. Predicting query performance. In *Proceedings of the 25th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 299–306.

Matthew Dahl, Varun Magesh, Mirac Suzgun, and Daniel E Ho. 2024. Large legal fictions: Profiling legal hallucinations in large language models. *Journal of Legal Analysis*, 16(1):64–93.

DeepSeek-AI. 2024. Deepseek-v3 technical report. *Preprint*, arXiv:2412.19437.

DeepSeek-AI, Daya Guo, Dejian Yang, Haowei Zhang, Junxiao Song, Ruoyu Zhang, Runxin Xu, Qihao Zhu, Shirong Ma, Peiyi Wang, Xiao Bi, Xiaokang Zhang, Xingkai Yu, Yu Wu, Z. F. Wu, Zhibin Gou, Zhihong Shao, Zhuoshu Li, Ziyi Gao, and 181 others. 2025. Deepseek-r1: Incentivizing reasoning capability in llms via reinforcement learning. *Preprint*, arXiv:2501.12948.

Mohamed Elaraby, Mengyin Lu, Jacob Dunn, Xueying Zhang, Yu Wang, Shizhu Liu, Pingchuan Tian, Yuping Wang, and Yuxuan Wang. 2023. Halo: Estimation and reduction of hallucinations in open-source weak large language models. *arXiv preprint arXiv:2308.11764*.

Farima Fatahi Bayat, Kun Qian, Benjamin Han, Yisi Sang, Anton Belyy, Samira Khorshidi, Fei Wu, Ihab Ilyas, and Yunyao Li. 2023. FLEEK: Factual error detection and correction with evidence retrieved from external knowledge. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 124–130, Singapore. Association for Computational Linguistics.

Team GLM, :, Aohan Zeng, Bin Xu, Bowen Wang, Chenhui Zhang, Da Yin, Dan Zhang, Diego Rojas, Guanyu Feng, Hanlin Zhao, Hanyu Lai, Hao Yu, Hongning Wang, Jiadai Sun, Jiajie Zhang, Jiale Cheng, Jiayi Gui, Jie Tang, and 40 others. 2024. Chatglm: A family of large language models from glm-130b to glm-4 all tools. *Preprint*, arXiv:2406.12793.

Aaron Grattafiori, Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Alex Vaughan, Amy Yang, Angela Fan, Anirudh Goyal, Anthony Hartshorn, Aobo Yang, Archi Mitra, Archie Sravankumar, Artem Korenev, Arthur Hinsvark, and 542 others. 2024. The llama 3 herd of models. *Preprint*, arXiv:2407.21783.

Yuzhe Gu, Ziwei Ji, Wenwei Zhang, Chengqi Lyu, Dahua Lin, and Kai Chen. 2024. Anah-v2: Scaling analytical hallucination annotation of large language models. *Advances in Neural Information Processing Systems*, 37:60012–60039.

Lei Huang, Weijiang Yu, Weitao Ma, Weihong Zhong, Zhangyin Feng, Haotian Wang, Qianglong Chen, Weihua Peng, Xiaocheng Feng, Bing Qin, and 1 others. 2023. A survey on hallucination in large language models: Principles, taxonomy, challenges, and open questions. *ACM Transactions on Information Systems*.

Soyeong Jeong, Jinheon Baek, Sukmin Cho, Sung Ju Hwang, and Jong Park. 2024. Adaptive-RAG: Learning to adapt retrieval-augmented large language models through question complexity. In *Proceedings of*

the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers), pages 7036–7050, Mexico City, Mexico. Association for Computational Linguistics.

Ziwei Ji, Yuzhe Gu, Wenwei Zhang, Chengqi Lyu, Dahua Lin, and Kai Chen. 2024. ANAH: Analytical annotation of hallucinations in large language models. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 8135–8158, Bangkok, Thailand. Association for Computational Linguistics.

Ziwei Ji, Nayeon Lee, Rita Frieske, Tiezheng Yu, Dan Su, Yan Xu, Etsuko Ishii, Ye Jin Bang, Andrea Madotto, and Pascale Fung. 2023. Survey of hallucination in natural language generation. *ACM Computing Surveys*, 55(12):1–38.

Ehsan Kamalloo, Nouha Dziri, Charles Clarke, and Davood Rafiei. 2023. Evaluating open-domain question answering in the era of large language models. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 5591–5606, Toronto, Canada. Association for Computational Linguistics.

Potsawee Manakul, Adian Liusie, and Mark Gales. 2023. SelfCheckGPT: Zero-resource black-box hallucination detection for generative large language models. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 9004–9017, Singapore. Association for Computational Linguistics.

Dasha Metropolitansky and Jonathan Larson. 2025. Towards effective extraction and evaluation of factual claims. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 6996–7045, Vienna, Austria. Association for Computational Linguistics.

Sewon Min, Kalpesh Krishna, Xinxi Lyu, Mike Lewis, Wen-tau Yih, Pang Koh, Mohit Iyyer, Luke Zettlemoyer, and Hannaneh Hajishirzi. 2023. FActScore: Fine-grained atomic evaluation of factual precision in long form text generation. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 12076–12100, Singapore. Association for Computational Linguistics.

Jingwei Ni, Minjing Shi, Dominik Stammbach, Mrinmaya Sachan, Elliott Ash, and Markus Leippold. 2024. AFaCTA: Assisting the annotation of factual claim detection with reliable LLM annotators. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1890–1912, Bangkok, Thailand. Association for Computational Linguistics.

OpenAI, Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, Red Avila, Igor Babuschkin, Suchir Balaji, Valerie Balcom, Paul Baltescu, Haiming Bao, Mohammad Bavarian, Jeff Belgum, and 262 others. 2024. Gpt-4 technical report. *Preprint*, arXiv:2303.08774.

Ankit Pal, Logesh Kumar Umapathi, and Malaikannan Sankarasubbu. 2023. Med-HALT: Medical domain hallucination test for large language models. In *Proceedings of the 27th Conference on Computational Natural Language Learning (CoNLL)*, pages 314–334, Singapore. Association for Computational Linguistics.

Yuehan Qin, Shawn Li, Yi Nian, Xinyan Velocity Yu, Yue Zhao, and Xuezhe Ma. 2025. Don't let it hallucinate: Premise verification via retrieval-augmented logical reasoning. *arXiv preprint arXiv:2504.06438*.

Qwen, :, An Yang, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chengyuan Li, Dayiheng Liu, Fei Huang, Haoran Wei, Huan Lin, Jian Yang, Jianhong Tu, Jianwei Zhang, Jianxin Yang, Jiaxi Yang, Jingren Zhou, and 25 others. 2025. Qwen2.5 technical report. *Preprint*, arXiv:2412.15115.

Nils Reimers and Iryna Gurevych. 2019. Sentence-bert: Sentence embeddings using siamese bert-networks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics.

Albert Sawczyn, Jakub Binkowski, Denis Janiak, Bogdan Gabrys, and Tomasz Kajdanowicz. 2025. Factselfcheck: Fact-level black-box hallucination detection for llms. *arXiv preprint arXiv:2503.17229*.

Hinrich Schütze, Christopher D Manning, and Prabhakar Raghavan. 2008. *Introduction to information retrieval*, volume 39. Cambridge University Press Cambridge.

Aaditya Singh, Adam Fry, Adam Perelman, Adam Tart, Adi Ganesh, Ahmed El-Kishky, Aidan McLaughlin, Aiden Low, AJ Ostrow, Akhila Ananthram, Akshay Nathan, Alan Luo, Alec Helyar, Aleksander Madry, Aleksandr Efremov, Aleksandra Spyra, Alex Baker-Whitcomb, Alex Beutel, Alex Karpenko, and 465 others. 2025. Openai gpt-5 system card. *Preprint*, arXiv:2601.03267.

Gemma Team, Aishwarya Kamath, Johan Ferret, Shreya Pathak, Nino Vieillard, Ramona Merhej, Sarah Perrin, Tatiana Matejovicova, Alexandre Ramé, Morgane Rivière, Louis Rouillard, Thomas Mesnard, Geoffrey Cideron, Jean bastien Grill, Sabela Ramos, Edouard Yvinec, Michelle Casbon, Etienne Pot, Ivo Penchev, and 197 others. 2025. Gemma 3 technical report. *Preprint*, arXiv:2503.19786.

Herbert Ullrich, Tomáš Mlynář, and Jan Drchal. 2025. Claim extraction for fact-checking: Data, models, and automated metrics. *arXiv preprint arXiv:2502.04955*.

Miriam Wanner, Seth Ebner, Zhengping Jiang, Mark Dredze, and Benjamin Van Durme. 2024. A closer look at claim decomposition. In *Proceedings of the 13th Joint Conference on Lexical and Computational Semantics (*SEM 2024)*, pages 153–175, Mexico City, Mexico. Association for Computational Linguistics.

Jerry Wei, Chengrun Yang, Xinying Song, Yifeng Lu, Nathan Hu, Jie Huang, Dustin Tran, Daiyi Peng, Ruibo Liu, Da Huang, and 1 others. 2024. Long-form factuality in large language models. *Advances in Neural Information Processing Systems*, 37:80756–80827.

An Yang, Anfeng Li, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Gao, Chengen Huang, Chenxu Lv, and 1 others. 2025. Qwen3 technical report. *arXiv preprint arXiv:2505.09388*.

Jiawei Zhang, Chejian Xu, Yu Gai, Freddy Lecue, Dawn Song, and Bo Li. 2024. Knowhalu: Hallucination detection via multi-form knowledge based factual checking. *arXiv preprint arXiv:2404.02935*.

Tianhang Zhang, Lin Qiu, Qipeng Guo, Cheng Deng, Yue Zhang, Zheng Zhang, Chenghu Zhou, Xinbing Wang, and Luoyi Fu. 2023a. Enhancing uncertainty-based hallucination detection with stronger focus. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 915–932, Singapore. Association for Computational Linguistics.

Yue Zhang, Yafu Li, Leyang Cui, Deng Cai, Lemao Liu, Tingchen Fu, Xinting Huang, Enbo Zhao, Yu Zhang, Yulong Chen, and 1 others. 2023b. Siren's song in the ai ocean: a survey on hallucination in large language models. *arXiv preprint arXiv:2309.01219*.

A Implementation of the JointCQ Framework

A.1 Data Generation

We sample 2,000 Chinese and 2,000 English questions from the ANAH-v2 (Gu et al., 2024) dataset. Answers are generated by four LLMs: `Qwen2.5-7B-Instruct` (Qwen et al., 2025), `Llama-3.1-8B-Instruct` (Grattafiori et al., 2024), `gemma-3-4b-it` (Team et al., 2025), `glm-9b-chat` (GLM et al., 2024). The prompt consists of only the question (without additional instructions) to simulate real-world usage. Detailed statistics are provided in Table 5.

Model	Qwen	Llama	Gemma	GLM	Total
N en	495	490	502	513	2000
N zh	646	0	687	667	2000
Total	1141	490	1189	1180	4000

Table 5: Statistics of generated answers in data sourcing stage.

We then synthesize claims and queries for QA pairs using few-shot prompting. The claim generation prompt is provided in Tables 13 and 14, while the query generation prompt is detailed in Tables 15 and 16. The generator's temperature is set to 0.9.

	Claim			Query		
Criterion	Ent.	Cov.	Dec.	Rel.	Con.	Usa.
Pass cnt	3,843	3,635	2,472	29,216	28,924	29,258
Pass rate(%)	96.08	90.88	61.80	99.23	98.23	99.37

Table 6: Statistics of data filtering.

A.2 Data Filtering

Prompt templates for claim and query filtering are shown in Tables 17 and 18 (claims) and Tables 19 and 20 (queries). Each evaluation assesses only one criterion at a time, with the evaluator's temperature set to 0.0 for maximum accuracy.

Initial filtering statistics (Table 6) reveal that decontextualization is the most challenging criterion, with an initial pass rate of 61.8%, while other criteria maintain pass rates above 90%. For samples failing the initial filter, we iteratively repeat the synthesis and filtering process until obtaining over 3,000 qualified samples for subsequent training data sampling.

B Implementation of Hallucination Detection Pipeline

The claim and query generation process uses the prompt templates shown in Tables 21 and 22. During the search stage, we configure the system to return 10 results per query. For verification, we employ the prompt templates in Tables 23 and 24. The same model generates outputs for both languages, with only the prompt templates differing. During postprocessing, responses labeled "Irrelevant" are automatically mapped to "Unverifiable". To minimize the influence of randomness, the temperature parameters of the model are uniformly set to 0.

C Experiments

C.1 Evaluation Dataset Statistics

Our evaluation is built upon test sets covering two languages, spanning multiple domains, and containing answers of varying lengths. We report detailed statistics on domain and answer length distributions to characterize the diversity of our evaluation benchmarks.

The domain distribution across all evaluation test sets is summarized in Table 7. ANAH focuses on four entity-centric domains, while HalluQA covers a broad spectrum of knowledge-intensive domains, ensuring robust and generalizable evaluation.

Table 8 shows the answer length distribution (in tokens) for all evaluation test sets, covering intervals from 0–15 to 512–1023 tokens. The test sets include both short and long-form answers, enabling a comprehensive evaluation across different response lengths.

Dataset	Domain (Count)
ANAH-en	Location (228), Person (201), Thing (52), Event (19)
ANAH-zh	Location (171), Person (133), Thing (101), Event (95)
HalluQA	History (28), Commonsense (24), Geography (21), Literature (19), Science (19), Classical Chinese (18), Poetry (16), Film and television (13), Music (11), Idiom (11), Art (10), Celebrity (8), Sports (5), Nature (3)

Table 7: Domain distribution of evaluation test sets.

N Tok.	ANAH-en	ANAH-zh	HalluQA
0–15	201	37	32
16–31	113	128	46
32–63	50	116	38
64–127	82	77	47
128–255	44	66	38
256–511	10	64	5
512–1023	0	12	0

Table 8: Answer length distribution of evaluation test sets.

C.2 Implementation of Baselines

We employ LLMs as baseline for our response-level evaluation. The hallucination detection prompts for these LLMs are provided in Tables 25 and 26, supporting only binary classification at the response level.

We configure SelfCheckGPT (Manakul et al., 2023) with a sample size of 20 and temperature of 1.0, computing consistency scores using the recommended NLI method.

For HaluAgent (Cheng et al., 2024) and FacTool (Chern et al., 2023), we utilize GPT-4.1 (OpenAI et al., 2024) through the GPT API for all external model calls and keep other inference parameters.

C.3 Results of Different Evaluation Settings

We propose an alternative evaluation approach that excludes unverifiable or failed samples, focusing solely on the verifiable portions. Notably, the composition of verifiable samples varies across different evaluation methods.

Response-level evaluation results are presented in Table 11. Sentence-level evaluation results are shown in Table 12.

Our method demonstrates consistent superiority over baseline approaches across both evaluation settings, maintaining robust performance.

D Annotation Protocol

D.1 Annotation for the Filtering Process

This section provides the annotation details for Section 3.3.4.

Qualification To ensure high-quality data labeling, we recruited a team of qualified annotators and established a rigorous annotation protocol. Given the bilingual nature of our dataset, all annotators were native Chinese speakers who had successfully passed the College English Test Band 6 (CET-6). This qualification ensured they possessed the necessary proficiency in both Chinese and English to accurately comprehend and process the multilanguage content.

Guidelines Prior to the formal annotation phase, annotators were provided with comprehensive written guidelines (Table 9) that detailed the research background and provided formal definitions for each evaluation dimension.

Compensation To maintain motivation and reflect the varying complexity of the tasks, we implemented a tiered compensation structure. Specifically, annotators were compensated at a rate of 2 RMB, 0.5 RMB, and 1.5 RMB per item for Subtasks 1, 2, and 3, respectively.

D.2 Annotation for the Verifier

To assess the reliability of the verifier, we manually annotate a set of claims and compare the verifier

Section	Content
Task Description	The evaluation consists of three subtasks: 1. Given a question, an answer, and a set of claims, evaluate the entailment and coverage of the claim set. 2. Given a single claim, evaluate its decontextualization property. 3. Given a claim and a query, evaluate the relevance, conciseness, and usability of the query.
Background	This is a QA-based annotation task where answers are generated by a large language model. A claim is a snippet extracted from the answer, defined as follows: *A claim is a statement explicitly presenting verifiable facts. Here, a fact is an assertion that can be objectively verified as true or false based on empirical evidence or reality.* Queries are used to assist in fact-checking the claims.
Operation	Fill in the blank "" for the corresponding dimension with 0 (does not meet the criterion) or 1 (meets the criterion).
Evaluation Criteria	
Entailment	The content of the claims should be fully supported by the source text, i,e, the question and answer. Examine each claim one by one to ensure every claim is supported.
Coverage	The extracted claims should capture all the verifiable factual information in the source text. Treat all claims as a whole and compare them with the QA pair to ensure coverage. *Note*: Subjective opinions, emotional expressions, and vague judgments do not need to be covered.
Decontextualization	The claim should be understandable on its own, without requiring additional context.
Relevance	The query directly relates to the claim, addressing its content, implications, or underlying assumptions.
Conciseness	The query should be clear and focused on the core information. Avoid multiple complex ideas or detailed descriptions in one query.
Usability	The query should use natural, fluent, and easily readable language that can yield relevant and accurate results from Google Search. Actual use of a search engine to verify query usability is encouraged.

Table 9: Annotation guideline.

model's predictions against these human-provided labels (Section 5.5). This section provides additional details about the annotation process. We recruit three volunteers familiar with the topic of hallucinations in LLMs. Each claim is independently annotated by one annotator. For each annotation, the annotator is provided with the claim and the corresponding retrieved documents. The annotation guidelines are consistent with the evaluation criteria presented in Tables 23 and 24. Annotators are informed that the dataset and the resulting annotations are used solely for research purposes.

E AI Usage Disclosure

In this work, we employ generative AI to support data analysis and enhance our manuscript. While using AI tools, we meticulously evaluate and edit the output to maintain the precision and credibility of our research.

F Ethical Considerations

We carefully consider the ethical aspects of our work on hallucination detection in general-domain question answering. All hallucinated contents in our datasets are explicitly labeled to ensure transparent and responsible use. We expect that the research poses minimal risks, as it does not involve sensitive data or human subjects. Our study uses only publicly available datasets and pretrained models that are licensed for academic use, and our use of these resources strictly follows their intended research purposes. The data we use do not contain any personally identifiable or sensitive information, and we assume that the original dataset providers perform appropriate anonymization and content filtering. The artifacts (datasets and models) developed in this work are released for research purposes only under terms consistent with the original licenses.

Criterion	Input QA/Claim	Failed	Passed
Entailment	Q: Where did Judas Iscariot lead the armed guard to arrest Jesus? A: In the Garden of Gethsemane.	The Latvian name for the Convent Yard is "Karmelītes parkāts."	Judas Iscariot led the armed guard to arrest Jesus in the Garden of Gethsemane.
Coverage	Q: When was the railway line beyond Dennington to Port Fairy closed? A: It was closed on 14 December 1982. This closure was part of a broader trend ...	The railway line from Dennington to Port Fairy in Victoria, Australia, was closed on 14 December 1982.	The railway line from Dennington to Port Fairy in Victoria, Australia, was closed on 14 December 1982. The closure of ... was part of a broader trend ...
Decontext.	Q: Which university library ... ? A: ... the Special Collections department has an archive dedicated to De Niro, which includes scripts ...	... The archive includes scripts ...	... The archive dedicated to Robert De Niro in the Special Collections department includes scripts ...
Relevance	Johann Strauss II is Mozart's father.	Who is Johann Strauss II's father?	Who is Mozart's father?
Conciseness	Vines and grapes represent the connection between Christ and the Eucharist, as well as the idea of spiritual growth and abundance.	What is the symbolism of vines and grapes in Christianity, particularly their connection to Christ, the Eucharist, spiritual growth, and abundance?	What is the symbolism of vines and grapes in the Christianity?
Usability	Comte asserted that reason is not a source of knowledge but a tool for understanding knowledge obtained through observation.	Auguste Comte reason knowledge source observation assertion	What was Comte's view on the role of reason in acquiring knowledge?

Table 10: Passed and failed examples of evaluation criteria. The criteria for claims are entailment, coverage, and decontextualization. The criteria for queries are relevance, conciseness, and usability.

	ANAH-en			ANAH-zh			ANAH-overall			HalluQA		
	Acc	F1	N	Acc	F1	N	Acc	F1	N	Acc	F1	N
FacTool	74.54	**78.01**	487	68.92	77.04	489	71.72	77.49	976	56.20	42.18	194
HaluAgent-13B	75.99	73.44	479	68.58	70.16	471	72.32	71.69	950	78.16	83.75	-
Ours	**76.36**	77.54	495	**73.62**	**78.61**	489	**75.00**	**78.11**	984	**82.09**	**84.08**	201

Table 11: Response-level evaluation results for the verifiable part. Accuracy (Acc) and F1 scores are reported as percentages. The results for HaluAgent-13B on the HalluQA dataset are sourced from (Cheng et al., 2024). Here, N denotes the number of samples used for metric calculation: ANAH contains 500 samples per language, while HalluQA consists of 206 samples.

	ANAH-en			ANAH-zh			ANAH-all		
	Acc	F1	N	Acc	F1	N	Acc	F1	N
FacTool	74.66	69.54	947	68.26	69.27	794	71.74	69.40	1741
Ours	**80.77**	**74.34**	905	**76.63**	**74.01**	736	**78.92**	**74.22**	1641
w/o filtering	78.16	70.81	902	76.48	73.81	727	77.41	72.29	1629
filter c only	79.38	72.46	907	74.21	71.01	725	77.08	71.15	1632
filter q only	79.91	72.78	901	76.38	73.07	724	78.34	72.92	1625
w/o q	78.79	70.68	896	73.94	69.75	729	76.62	70.22	1625
replace CQG	76.40	69.02	894	75.57	73.12	704	76.03	71.01	1598
w/o joint	78.91	70.89	915	75.87	73.72	721	77.57	72.30	1636

Table 12: Sentence-level hallucination detection results for the verifiable part of the ANAH dataset. The evaluation covers 1,037 English sentences and 839 Chinese sentences.

English Prompt Template of Claim Synthesis

Task
Given a pair of question and answer as input, your task is to extract all claims.

Task Rules
When extracting claims, strictly follow these rules:
1. Claims must be factual statements that can be verified or refuted. Exclude subjective opinions, emotional expressions, and vague judgments.
2. Each claim must be semantically complete and independently understandable without relying on context.
3. Avoid ambiguous pronouns in claims. Use specific nouns for clarity.
4. Extract and output all qualifying claims, with each claim on a separate line.
5. If no claims meeting the above criteria exist in the input, output "No claims."
6. Strictly follow the specified format in the response, without adding extra explanations or unrelated content.

Examples
{examples}

Input
[Question]
{question}
[Answer]
{answer}
[Claims]

Table 13: English prompt template of claim synthesis.

Chinese Prompt Template of Claim Synthesis

任务
给定一对问题和回答作为输入，你的任务是提取所有的陈述。

任务规则
提取陈述时请严格遵循以下规则：
1. 陈述必须是可以核实或驳斥的事实性声明。排除主观意见、情绪表达和模糊判断。
2. 每条陈述必须语义完整，不依赖上下文即可独立理解其含义。
3. 陈述中禁止使用指代不明的代词，必须使用具体名词表述。
4. 必须提取并输出所有符合条件的陈述，每条陈述独占一行。
5. 当输入中不存在符合上述标准的陈述时，输出"无陈述"。
6. 必须严格按照指定格式回复，不得添加其他内容，不得添加多余的解释说明。

示例
{examples}

输入
[问题]
{question}
[回答]
{answer}
[陈述]

Table 14: Chinese prompt template of claim synthesis.

English Prompt Template of Query Synthesis

Task
Given a claim, your task is to generate a search engine query to help fact-check the claim.

Task Rules
When generating the query, strictly follow these rules:
1. The query should be concise and clear, specifically targeting the claim to be verified.
2. The query should be applicable to search engines and can help users obtain valid information.
3. Always output a query.
4. If there is nothing to query, output "No query".
5. You must strictly follow the specified format. Do not add any extra content or explanations.

Examples
{examples}

Input
[Claim]
{claim}
[Query]

Table 15: English prompt template of query synthesis.

Chinese Prompt Template of Query Synthesis

任务
给定一条陈述，你的任务是生成一条搜索引擎查询，用于协助对该陈述进行事实核查。

任务规则
生成查询时请严格遵循以下规则：
1. 查询应当简洁明确，对待验证的陈述具有针对性。
2. 查询能够应用于搜索引擎的搜索，帮助用户获取有效信息。
3. 始终输出一条查询语句。
4. 若无待查询的内容，直接输出"无查询"。
5. 必须严格按照指定格式回复，不得添加其他内容，不得添加多余的解释说明。

示例
{examples}

输入
[陈述]
{claim}
[查询]

Table 16: Chinese prompt template of query synthesis.

English Prompt Template of Claim Filtering

Task
You are provided with a question, its answer, a set of claims (a claim) extracted from the QA pair. Your task is to assess whether the claim(s) satisfy the specific criterion.

Evaluation Criteria
The claim(s) should meet the following criterion:
Entailment: The content of the claims should be fully supported by the source text. Review each statement point by point to ensure that every statement is fully supported.
OR
Coverage: The extracted claims should capture all the verifiable factual information in the source text. Evaluate all claims collectively against the question and answer to verify full coverage.
OR
Decontextualization: The claim should be understandable on its own, without requiring additional context.

If the claim(s) meet the criterion, respond with "Yes"; otherwise, respond with "No".

Input
[Question]
{question}
[Answer]
{answer}
[Claim(s)]
{claims}

Table 17: English prompt template of claim filtering.

Chinese Prompt Template of Claim Filtering

任务
给定一个问题、其答案、一组(条)从问答对中提取的陈述，你的任务是评估这些(条)陈述是否满足特定的标准。

评估标准
陈述应当满足以下标准：
蕴含性：陈述的内容应完全由原文支持。逐条展开检查陈述，确保每条陈述都能被支持。
OR
覆盖性：提取出的这组陈述应涵盖原文中所有可验证的事实信息。视所有陈述为一个整体并与问答进行比较以确保覆盖性。
OR
去上下文化：每条陈述应在不需要额外上下文的情况下可以被理解。

如果这些(条)陈述符合标准，请回答"是"；否则，请回答"否"。

输入
[问题]
{question}
[回答]
{answer}
[陈述]
{claims}

Table 18: Chinese prompt template of claim filtering.

English Prompt Template of Query Filtering

Task
You are given a claim and a query intended for Google Search. Your task is to evaluate whether the query satisfies the specific criterion.

Evaluation Criteria
The query is considered helpful if it meets the following criterion:
Relevance: The query directly relates to the claim, addressing its content, implications, or underlying assumptions.
OR
Conciseness: The query should be clear and focused on the core information, avoiding multiple complex ideas or detailed descriptions in one query.
OR
Usability: The query should use natural, fluent, and easily readable language that can yield relevant and accurate results from Google Search.

If the query meets the criterion, respond with "Yes"; otherwise, respond with "No". No additional explanation is allowed.

Input
[Claim]
{claim}
[Query]
{query}

Table 19: English prompt template of Query filtering.

Chinese Prompt Template of Query Filtering

任务
给定一条陈述和一条用于Google搜索的查询，你的任务是评估该查询是否满足特定的标准。

评估标准
如果查询符合以下标准，则认为它是有帮助的：
相关性：提问需紧扣陈述本身，涉及其内容、含义或背后的假设。
OR
简洁性：查询应简明扼要，聚焦核心信息，避免在一个查询中包含多个复杂概念或细节描述。
OR
可用性：查询应使用自然、流畅且易读的语言，以便从Google搜索中获得相关且准确的结果。

如果查询满足以上标准，请回答"是"；否则，请回答"否"。不允许输出任何额外解释。

输入
[陈述]
{claim}
[查询]
{query}

Table 20: Chinese prompt template of query filtering.

English Prompt and Response Templates of Claim-Query Generator

Task
Given a question and an answer as input, your task is to extract all claims, and generate a search engine query for each claim to help fact-check the claims.

Task Rules
When extracting claims, strictly follow these rules:
1. Claims must be factual statements that can be verified or refuted. Exclude subjective opinions, emotional expressions, and vague judgments.
2. Each claim must be semantically complete and independently understandable without relying on context.
3. Avoid ambiguous pronouns in claims. Use specific nouns for clarity.
4. Extract and output all qualifying claims, with each claim on a separate line.
5. If no claims meeting the above criteria exist in the input, output "No claims."

When generating the queries, strictly follow these rules:
1. The queries should be concise and clear, specifically targeting the claims to be verified.
2. The queries should be applicable to search engines and can help users obtain valid information.
3. If there is nothing to query, output "No query".

Input
[Question]
{question}
[Answer]
{answer}

[Claims]
{claims}
[Queries]
{queries}
[End]

Table 21: English prompt and response templates of Claim-Query Generator.

Chinese Prompt and Response Templates of Claim-Query Generator

任务
给定问题和回答作为输入，你的任务是提取所有的陈述，然后为每条陈述生成一条搜索引擎查询，用于协助对陈述进行事实核查。

任务规则
提取陈述时请严格遵循以下规则：
1. 陈述必须是可以核实或驳斥的事实性声明。排除主观意见、情绪表达和模糊判断。
2. 每条陈述必须语义完整，不依赖上下文即可独立理解其含义。
3. 陈述中禁止使用指代不明的代词，必须使用具体名词表述。
4. 必须提取并输出所有符合条件的陈述，每条陈述独占一行。
5. 当输入中不存在符合上述标准的陈述时，输出"无陈述"。

生成查询时请严格遵循以下规则：
1. 查询应当简洁明确，对待验证的陈述具有针对性。
2. 查询能够应用于搜索引擎的搜索，帮助用户获取有效信息。
3. 若无待查询的内容，直接输出"无查询"。

输入
[问题]
{question}
[回答]
{answer}

[陈述]
{claims}
[查询]
{queries}
[结束]

Table 22: Chinese prompt and response templates of Claim-Query Generator.

English Prompt Template of Verifier

Task
Given a claim and related reference searched by a query as input, your task is to determine whether the claim is valid based on the reference.

Evaluation Criteria
Please make your judgment based on the following criteria and choose one of the three options:
1. Correct: The reference supports the claim.
2. Hallucination: The reference is relevant to the claim, but does not support the claim.
3. Irrelevant: The reference is irrelevant to the claim, thus does not contain enough information to determine the factuality of the claim. Only use this option when absolutely necessary.

Provide only one option as the output. No additional explanation is allowed.

Input
[Claim]
{claim}
[Reference]
{reference}

Table 23: English prompt template of Verifier.

Chinese Prompt Template of Verifier

任务
给定一条陈述以及由查询检索到相关的参考资料作为输入，你的任务是根据参考资料判断陈述是否成立。

判断标准
请依据以下标准进行判断，输出三个选项之一：
1. 正确：参考资料能够支持陈述。
2. 幻觉：参考资料与陈述相关，但并不支持陈述。
3. 无关：参考资料与陈述内容无关，信息不足，无法判断陈述的真实性。非必要不使用此选项。

只输出一个选项，不允许有任何额外解释。

输入
[陈述]
{claim}
[参考资料]
{reference}

Table 24: Chinese prompt template of Verifier.

English Prompt Template of LLM Baselines

I want you act as an answer judge. Given a question and an answer, your objective is to determine if the provided answer contains non-factual or hallucinated information.
The response you give should be “Yes” or “No”.

Question
{question}
Answer
{answer}

Table 25: English prompt template of LLM baselines.

Chinese Prompt Template of LLM Baselines
我希望你充当一个答案评审员。给定一个问题和一个答案，你的任务是判断所提供的答案中是否包含虚假或幻觉的信息。 你的回复应该为“是”或“否”。 ### 问题 {question} ### 答案 {answer}

Table 26: Chinese prompt template of LLM baselines.