EVICheck: Evidence-Driven Independent Reasoning and Combined Verification Method for Fact-Checking

Lingxiao Wang[1] , Lei Shi[1]* , Feifei Kou[2,3] , Ligu Zhu[1] , Chen Ma[4] , Pengfei Zhang[5] , Mingying Xu[6] , Zeyu Li[1]

[1]State Key Laboratory of Media Convergence and Communication, Communication University of China
[2]School of Computer Science (National Pilot School of Software Engineering), BUPT
[3]Key Laboratory of Trustworthy Distributed Computing and Service, BUPT Ministry of Education
[4]Institute of Cyberspace Security, Zhejiang University of Technology
[5]State Key Laboratory of Digital Intelligent Technology for Unmanned Coal Mining, the School of Computer Science and Engineering, Anhui University of Science and Technology
[6]School of Artificial Intelligence and Computer Science, North China University of Technology
{wlx, leiky_shi}@cuc.edu.cn

Abstract

Large Language Models (LLMs) and Retrieval-Augmented Generation (RAG) have demonstrated significant potential in automated fact-checking. However, existing methods face limitations in insufficient evidence utilization and lack of explicit verification criteria. Specifically, these approaches aggregate evidence for collective reasoning without independently analyzing each piece, hindering their ability to leverage the available information thoroughly. Additionally, they rely on simple prompts or few-shot learning for verification, which makes truthfulness judgments less reliable, especially for complex claims. To address these limitations, we propose a novel method to enhance evidence utilization and introduce explicit verification criteria, named EVICheck. Our approach independently reasons each evidence piece and synthesizes the results to enable more thorough exploration and enhance interpretability. Additionally, by incorporating fine-grained truthfulness criteria, we make the model's verification process more structured and reliable, especially when handling complex claims. Experimental results on the public RAWFC dataset demonstrate that EVICheck achieves state-of-the-art performance across all evaluation metrics. Our method demonstrates strong potential in fake news verification, significantly improving the accuracy.

1 Introduction

Fact-checking involves verifying the accuracy of claims or information, often to determine whether they are true or false. Traditionally, experts manually assess claims using authoritative sources and their expertise, with platforms like PolitiFact[1] leading the way. Automated approaches, such as FEVER [Thorne *et al.*, 2018], scale the verification process using knowledge bases like Wikipedia. However, manual methods are limited by scale, and automated techniques, while scalable, struggle with complex claims and with ensuring verification accuracy without human oversight.

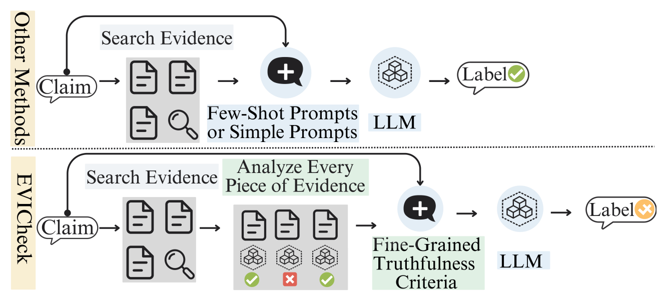

Figure 1: Comparison between EVICheck and other methods: other methods perform reasoning based on simple prompts after collecting evidence, while EVICheck independently analyzes each piece of evidence and decides based on fine-grained truthfulness criteria.

Recent advancements in Large Language Models (LLMs) and Retrieval-Augmented Generation (RAG) have improved automated fact-checking systems [Guu *et al.*, 2020; Izacard and Grave, 2021]. With strong language comprehension and efficient use of external knowledge bases, these systems offer notable advantages in tackling fact-checking tasks [Ostrowski *et al.*, 2021; Chen *et al.*, 2024; Lewis *et al.*, 2020]. However, existing methods face persistent challenges in verifying complex disinformation [Chern *et al.*, 2023; Khaliq *et al.*, 2024]. One major limitation lies in evidence utilization [Atanasova *et al.*, 2020; Kotonya and Toni, 2020]. While evidence is often collected through various approaches, most methods rely on overall validation instead of independently analyzing each piece of evidence, resulting in incomplete exploration of available information [Khaliq *et al.*, 2024; Yue *et al.*, 2024]. We term this problem Insufficient Evidence Utilization (IEU). Furthermore, many methods lack clear verification standards, often using simple prompts or

*Corresponding author.
[1]https://www.politifact.com/

few-shot examples for credibility verification, which undermines their reliability when dealing with complex or ambiguous claims [Zhang and Gao, 2023; Yue *et al.*, 2024]. Addressing these gaps requires more effective evidence utilization and the establishment of robust verification standards.

Inspired by structured reasoning and evidence integration used by humans to address complex issues, recent research has sought to improve the reliability and transparency of automated systems [Chern *et al.*, 2023; Zamani and Bendersky, 2024]. However, many approaches struggle to integrate evidence verification and reasoning effectively, limiting their capacity to process complex information [Khaliq *et al.*, 2024; Zhang and Gao, 2023]. To overcome these challenges, we propose EVICheck, a novel method that enhances evidence-based reasoning and introduces fine-grained truthfulness criteria to improve fact-checking performance.

EVICheck performs independent reasoning for each piece of evidence, ensuring that each piece is thoroughly analyzed and utilized rather than being overshadowed by collective aggregation, as shown in Figure 1. It also introduces fine-grained truthfulness criteria, making the evaluation process more structured and reliable. We also integrate search engine APIs (e.g., SerpApi[2]) into the RAG process to ensure the system can access the most up-to-date relevant information, enhancing the model's real-time capability and accuracy. Experimental results show that EVICheck performs best on the public RAWFC dataset, particularly demonstrating superior potential compared to existing methods when handling complex, ambiguous, or controversial statements.

Our contributions can be summarized as follows:

- We propose the EVICheck method, which fully utilizes each piece of evidence by performing independent reasoning, enhancing the comprehensiveness and effectiveness of evidence utilization.
- We construct fine-grained truthfulness criteria within the framework, offering a more structured and reliable fact-checking process and improving the model's decision-making accuracy, particularly on complex statements.
- We demonstrate the superiority of EVICheck through experiments on the public RAWFC dataset, showing its significant application potential in fact-checking.

2 Related Work

Fact-Checking Based on LLMs and RAG. Large Language Models (LLMs) and Retrieval-Augmented Generation (RAG) methods are effective tools for improving fact-checking accuracy. While LLMs embed extensive knowledge through pretraining, relying on static data limits their timeliness and knowledge coverage [Izacard and Grave, 2021; Wang *et al.*, 2023]. RAG methods address this by incorporating external knowledge sources to enhance LLMs' knowledge acquisition [Borgeaud *et al.*, 2022; Wu *et al.*, 2022]. For example, LLM-Augmenter [Peng *et al.*, 2023] combines local knowledge bases with automatic feedback. However, its performance is constrained by delays in updating the knowledge base. To improve timeliness, researchers have utilized search engine APIs, such as RAGAR [Khaliq *et al.*, 2024] with the DuckDuckGo Search. However, these methods often focus on aggregating multiple pieces of evidence into a single decision-making process without independently analyzing each piece, leading to insufficient utilization of the available evidence. This limits their ability to explore and validate contributions of evidence in fact-checking tasks fully.

Evidence-Based Interpretable Fact-Checking. Evidence-based fact-checking methods verify claims by retrieving relevant evidence, typically using external knowledge sources like knowledge graphs or document fragments [Shang *et al.*, 2022; Wang and Shu, 2023; Zhao *et al.*, 2023]. With the advent of LLMs (e.g., GPT), generative methods have gained traction in providing transparent, human-like explanations. These methods combine extractive and abstractive summarization to extract key information and generate coherent, contextually connected explanations, thereby improving interpretability [Atanasova *et al.*, 2020; Kotonya and Toni, 2020]. However, most existing methods lack fine-grained truthfulness criteria to evaluate claims systematically. This absence of structured guidance limits the reliability of these methods, particularly when dealing with complex or ambiguous claims where precise decision-making is crucial.

In summary, fact-checking methods using LLMs and RAG improve accuracy and interpretability by integrating external knowledge and generating human-like explanations. However, they struggle with insufficient evidence utilization, fine-grained truthfulness analysis, and timely updates. To bridge these gaps, we propose a novel method that optimizes evidence utilization and refines verification criteria for greater accuracy and robustness in real-world applications.

3 Our EVICheck Method

Task Definition. The task is to perform automated fact-checking of claims by evaluating their truthfulness using a multi-step reasoning process. Given a claim x, the goal is to determine its validity by retrieving relevant evidence, performing reasoning, and providing a conclusion along with an explanation. This task can be formally defined as follows:

$$(\hat{y}, e) = f_{\text{validate}}(x), \tag{1}$$

where x is the input claim to be validated, $\hat{y}$ is the truthfulness judgment (True, False, or Half), and e is the corresponding explanation. The function $f_{\text{validate}}(x)$ encompasses all the steps, including question generation, evidence retrieval, reasoning, and final aggregation of results.

As shown in Figure 2, the EVICheck method has two main modules: evidence acquisition with preliminary reasoning and combined verification based on fine-grained truthfulness criteria. The first module follows a four-step loop: generating verification questions, selecting the best one, retrieving relevant information for preliminary reasoning, and generating new questions to gather additional evidence. This process ensures comprehensive evidence collection. The second module, combined verification, integrates the collected evidence and uses fine-grained truthfulness criteria for structured evaluation, enabling accurate decision-making. The algorithm of our method is shown in Algorithm 1.

[2]https://serpapi.com/manage-api-key

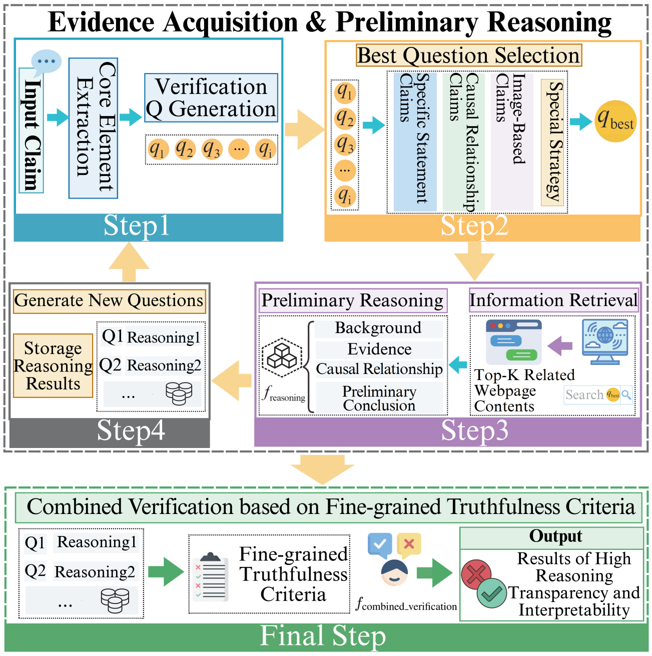

Figure 2: Retrieval-augmented fact verification with multi-round inference and combined verification. Evidence acquisition and preliminary reasoning use GPT and a search-engine API; after the final reasoning loop, all results are fed into the fine-tuned model using fine-grained truthfulness criteria for combined verification.

3.1 Evidence Acquisition & Preliminary Reasoning

Element Extraction and Question Generation

In this step, we aim to identify the core elements of the claim and generate verification questions from multiple aspects of the claim, as shown in Step 1 of Figure 2. For claim x, we first extract core elements, identifying key facts and verifiable information. This step ensures that the subsequent verification questions $\{q_i\}_i^m$ are focused on the aspects of the claim that require validation, preventing the process from diverging to irrelevant details. Based on these core elements, we generate targeted verification questions from multiple perspectives, ensuring alignment with the claim's content and verifiability through reliable evidence (e.g., official documents, news reports, or eyewitness accounts).

Best Question Selection

In this step, the goal is to select the most relevant validation question from the set of generated questions for retrieval, ensuring that the question chosen aligns closely with the claim's context and is effective for fact-checking, as shown in Step 2 of Figure 2. Once validation questions are generated, the large language model f_{select} selects the most relevant question q_{best} from the set $\{q_i\}_i^m$ for retrieval. The model assigns a relevance score to each question based on predefined evaluation criteria, selecting the question with the highest score:

$$q_{\text{best}} = \arg \max_{i \in \{1,\ldots,m\}} f_{\text{select}}(q_i, x), \tag{2}$$

where $\arg\max$ selects the question q_i with the highest score $f_{\text{select}}(q_i, x)$, i.e., the most relevant question to the claim x.

The rationale behind choosing the best question includes reducing noise, improving efficiency, and addressing context length limitations. By choosing the most relevant question, irrelevant or noisy queries are avoided, which enhances the accuracy of verification and reduces unnecessary API calls. This approach conserves computational resources and improves overall efficiency. Additionally, pre-selecting the best question ensures the context window stays within its limits, allowing the model to process all relevant information.

Furthermore, in practice, we observed the following deficiencies in LLMs when selecting optimal questions: First, in causal relationship claims, the models tend to focus on the veracity of events A and B while neglecting the critical importance of their causal linkage. Second, when handling specific claims, the models tend to rely on official sources for evidence retrieval but struggle to obtain useful information for statements that are informally recorded or released through unofficial channels. Finally, for claims involving multimodal information such as images or videos, the models attempt to retrieve the corresponding multimodal data directly but often fail. To address these issues, we designed a series of special prompt strategies to assist the models in more accurately selecting and generating appropriate verification questions.

Information Retrieval & Preliminary Reasoning

In this step, the objective is to retrieve relevant information from external sources based on the selected question and then conduct preliminary reasoning to form an initial verification, as shown in Step 3 of Figure 2. After selecting the optimal question q_{best}, the search engine API is used to retrieve web content $\{w_i\}_{i=1}^n$ related to the selected question. Subsequently, the model $f_{\text{reasoning}}$ is used to perform integrated reasoning and summary analysis on the retrieved web content, generating a preliminary conclusion $\hat{y}_1$ and reasoning process e_1:

$$(\hat{y}_1, e_1) = f_{\text{reasoning}}(\{w_i\}_{i=1}^n,\ q_{\text{best}},\ x). \tag{3}$$

Each retrieved piece of information is analyzed individually, extracting key evidence to make a preliminary judgment. The overall preliminary reasoning result is then presented in a structured format containing the following elements:

- Background Information: Provides context to aid in understanding the background of the statement.
- Evidence: Lists key information and data extracted from each source.
- Causal Relationship: Analyzes the causal logic of the statement, assessing its rationality and consistency.
- Conclusion: Makes a preliminary verification regarding the truthfulness based on the analysis and evidence.

Loop Inference and Validation

In this step, the goal is to refine the verification process by repeatedly generating new validation questions based on the claim and the current evidence and reasoning, as shown in Step 4 of Figure 2. After the preliminary inference, new verification questions $\{q'_i\}_i^m$ are generated by combining the statement x with the previous inference results $\hat{y}_{i-1}$ and e_{i-1}. The generation of new questions is centered around the claim

Algorithm 1 Multi-Round Reasoning and Validation

Input: claim x, the number of iterations max_loops.
Output: Final Prediction $\hat{y}$, Explanation e.
$\{q_i\}_i^m \leftarrow \text{GenerateQuestions}(x)$
$q_{\text{best}} \leftarrow \text{SelectBestQuestion}(\{q_i\}_i^m, x)$
$\text{EvidenceSet} \leftarrow []$
$\text{counter} \leftarrow 0$
while counter $<$ max_loops **do**
  $w \leftarrow \text{RetrieveWebContent}(q_{\text{best}})$
  $\hat{y}_{\text{current}}, e_{\text{current}} \leftarrow \text{Reasoning}(w, q_{\text{best}}, x)$
  $\text{EvidenceSet.append}(\{\hat{y}_{\text{current}}, e_{\text{current}}\})$
  $\{q'_i\}_i^m \leftarrow \text{GenFollowQ}(\hat{y}_{\text{current}}, e_{\text{current}}, x)$
  $q_{\text{best}} \leftarrow \text{SelectBestQuestion}(\{q'_i\}_i^m, x)$
  $\text{counter} \leftarrow \text{counter} + 1$
end while
$(\hat{y}, e) \leftarrow \text{CombinedValidate}(\text{EvidenceSet}, \mathcal{S})$
return $\hat{y}, e$

rather than being entirely dependent on the questions or reasoning results from the previous round. This strategy avoids the issue of subsequent reasoning deviating from the core due to an initial question that is off-target. Each round of question generation ensures comprehensive validation by considering both the statement's key elements and prior reasoning results.

Once the loop counter reaches `max_loops`, the loop terminates and proceeds to the next module to start the final step.

3.2 Evidence Aggregation and Combined Verification

In this step, the goal is to aggregate the evidence from multiple rounds of reasoning and validation and then combine them to make a final, accurate verification, as shown in the Final Step of Figure 2. The process begins with several rounds of reasoning, where question-reasoning result pairs $\{(\hat{y}_i, e_i)\}_{i=1}^m$, which include validation questions and their corresponding background information, evidence, causal analysis, and preliminary conclusions—are generated and iteratively refined. After multiple rounds, these evidence and preliminary judgments are aggregated and validated, guided by a set of fine-grained truthfulness criteria $\mathcal{S}$, which define evaluation rules for conclusions (True, False, Half) and provide clear guidelines for the model, as shown in Table 1. The conclusion $\hat{y}$ and explanation e are derived, ensuring both high credibility and accuracy.

To further improve the accuracy of the model's verification, we fine-tuned the model $f_{\text{combined\_verification}}$ using a supervised learning approach on the X_{train} dataset. In the fine-tuning process, we used the LLaMA-Factory framework[3] [Zheng *et al.*, 2024] along with the LoRA (Low-Rank Adaptation) method [Hu *et al.*, 2022], freezing all parameters except for the LoRA adapters. The fine-tuning was performed for three epochs. The objective is to maximize the accuracy of the final prediction by minimizing the loss function, which quantifies the difference between predicted and actual outcomes:

$$\min_{\theta} \mathbb{E}_{(x,\hat{y}_i,e_i)\sim X_{\text{train}}} \left(L(\theta, x, \{\hat{y}_i, e_i\}_{i=1}^m, \mathcal{S}) \right), \quad (4)$$

[3]https://github.com/hiyouga/LLaMA-Factory/tree/main

TRUE	HALF	FALSE
Criteria: The claim fully aligns with the facts, there is sufficient evidence supporting it, and it accurately describes the actual situation. **Key Elements:** - The facts in the claim are fully verified, and all information is accurate and complete, without misleading elements or key facts missing. **Additional Notes:** - Do not overly rely on official documents; non-official sources (such as social media, eyewitness statements, etc.) can be used as evidence if they are sufficient and reliable. - In the absence of official statements, the claim can still be rated as "true" if other reliable sources sufficiently support its authenticity.	**Criteria:** The claim is partially correct but there is a significant omission of important information or neglect of context, making it potentially misleading or not fully accurate. **Key Elements:** - Partially Correct, Core Still Needs Verification: Some parts of the claim are correct, but not enough background or key information is provided, which leads to incomplete understanding or potential misunderstanding. - Core Correct: If the core of the claim is correct, even if other parts are incomplete or inaccurate, it can still be rated as half. - Misleading Details: The claim might simplify or exaggerate certain facts, which can lead to an incomplete or misleading understanding. - Language Details: Pay close attention to the language used; check if there are omissions or nuances in how the claim is stated that could affect the overall accuracy.	**Criteria:** The claim is clearly inconsistent with the facts, the evidence shows it contains incorrect information, misleading conclusions, or is inconsistent with the actual situation. **Key Elements:** - Causal Relationship Error: If the claim involves a causal relationship that is incorrect or cannot be established, it should be rated as false. - Evidence Does Not Support the Claim: If the evidence provided does not clearly support the claim, or if the evidence is completely inconsistent, the claim should be rated as false. - Unverified Claim: If the core of the claim has not been conclusively verified or the evidence is insufficient, the claim should be rated as false. - Official Direct Denial: If there is a direct denial from a reliable official source, the claim should be rated as false. - Lack of Key Information: If the core information of the claim cannot be verified or lacks clear evidence, it should be rated as false.

Table 1: Fine-grained truthfulness criteria.

	false	half	true	total
train	514	537	561	1612
test	66	67	67	200
validation	66	67	67	200

Table 2: RAWFC data statistics.

where θ represents the model parameters, x is the input statement, $\hat{y}_i$ and e_i are the predicted conclusions and explanations from the previous reasoning rounds, X_{train} is the training dataset, and L is the loss function that quantifies the difference between the model's prediction and the ground truth. The goal is to minimize the loss, ensuring that the model's final prediction $\hat{y}$ is as accurate as possible based on the aggregated evidence and the criteria $\mathcal{S}$.

4 Experimental Setting

Dataset. We adopt the English fake news dataset RAWFC [Yang *et al.*, 2022] for experiments. The dataset was created by collecting claims from Snopes[4] and retrieving the relevant raw reports. It includes three categories of labels: True, False, and Half, with each data entry provided with a manually annotated "golden label" explanation. The data distribution is shown in Table 2.

Experimental Details. The experiments were conducted using three different models: GPT-3.5, GPT-4, and the Llama-3-8B-Instruct model[5]. To make more accurate verification we fine-tuned Llama-3-8B-Instruct model as $f_{\text{combined\_verification}}$. SerpApi was used as the search API in the experiment. To reduce computational overhead was conducted based on $M = 5$ validation questions and performing $N = 2$ rounds of loop inference. Although we found that increasing the number of training rounds and questions could lead to improved results, further details are shown in Figure 5.

Baseline. We employ two categories of baselines:

- **Supervised methods**
  1) GenFE [Atanasova *et al.*, 2020]: multi-task explanation generation.
  2) SentHAN [Ma *et al.*, 2019]: hierarchical attention over sentence-level evidence.

[4]https://www.snopes.com/
[5]https://huggingface.co/meta-llama/Meta-Llama-3-8B-Instruct

3) SBERT [Kotonya and Toni, 2020]: interpretable fact-checking for public health claims.
4) CofCED [Yang *et al.*, 2022]: cascading evidence distillation for report-based detection.

- **LLM-based methods (GPT-3.5)**
  1) CoT [Wei *et al.*, 2022]: chain-of-thought prompting for complex reasoning.
  2) Standard Prompt [Brown *et al.*, 2020]: few-shot GPT-3 prompting.
  3) Hiss [Zhang and Gao, 2023]: hierarchical hinting for statement breakdown.
  4) RAFTS [Yue *et al.*, 2024]: retrieve-and-compare using LLM synthesis.

Evaluation Metrics. To comprehensively evaluate the model's performance, three evaluation metrics were used: Macro-average Precision (P), Recall (R), and F1-score. In addition to the metrics above, we used the confusion matrix to analyze further the model's performance in the three-class classification task. The matrix displays the relationship between true labels and predicted labels, helping identify misclassification patterns. By analyzing it, we gain insights into the model's performance, particularly in recognizing categories with significant misclassification issues, guiding future optimizations, as shown in Figure 3.

5 Experimental Results

This section presents the experimental results of the proposed method, starting with an evaluation of overall performance, including comparisons with existing approaches. Ablation experiments assess the impact of individual components on performance, while manual evaluation confirms the method's practical effectiveness. Finally, a detailed analysis of the validation process and a typical case is provided.

5.1 Performance Outcome

We conducted experiments separately using GPT-3.5 and GPT-4, and then replaced the $f_{\text{combined\_verification}}$ model with the fine-tuned Llama-3-8B-Instruct model for further experimentation. The experimental results of our method, comparing them with the current SOTA results, are shown in Table 3.

The experimental results demonstrate that under the conditions of using GPT-3.5 and fine-tuned Llama 3, our approach outperformed traditional SOTA methods across all evaluation metrics. When using GPT-4 and fine-tuned Llama 3, compared to traditional SOTA methods, our method improved accuracy by 3.5%, precision by approximately 8.6%, and F1 score by about 6.0%. These results indicate a significant improvement in the accuracy and reliability of verification in the fact-checking task. The observed performance improvements can be attributed to several key innovations in our approach. First, we incorporated a preliminary reasoning step after each information retrieval, ensuring that every piece of evidence is fully utilized. Second, we introduced a set of fine-grained truthfulness criteria to guide the model in making final verification, which enhanced its performance in determining the veracity of statements. Finally, we fine-tuned the Llama 3 model specifically to perform better in fact-checking tasks, optimizing its ability to discern factual accuracy. Together, these innovations contributed to the significant performance gains observed in our experiments.

Model	P	R	F1↑
Supervised Approaches			
GenFE	0.443	0.448	0.445
SentHAN	0.457	0.455	0.456
SBERT	0.511	0.460	0.484
CofCED	0.530	0.510	0.520
Methods with GPT-3.5			
CoT	0.424	0.466	0.444
Standard Prompt	0.485	0.485	0.485
Hiss	0.534	<u>0.544</u>	0.539
RAFTS	<u>0.628</u>	0.526	<u>0.573</u>
Ours			
EVICheck (w/ GPT-3.5)	0.577	0.580	0.579
EVICheck (w/ GPT-3.5 + Llama $3_{\text{fine-tuned}}$)	0.630	0.615	0.619
EVICheck (w/ GPT-4)	0.645	0.600	0.584
EVICheck (w/ GPT-4 + Llama $3_{\text{fine-tuned}}$)	**0.663**	**0.630**	**0.633**

Table 3: Experimental results of claim verification. Supervised results are from [Yang *et al.*, 2022]; Standard Prompt and CoT results are from [Zhang and Gao, 2023]. Llama $3_{\text{fine-tuned}}$ denotes the fine-tuned Llama-3-8B-Instruct model.

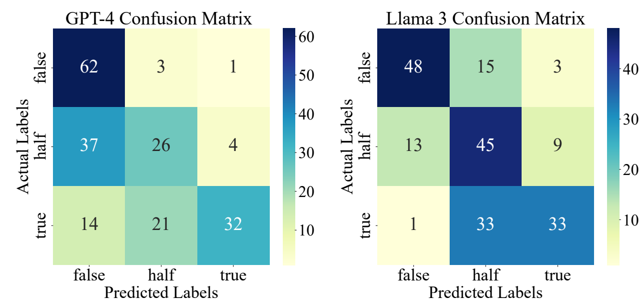

Figure 3: Confusion matrix heatmap. Left: GPT-4. Right: the fine-tuned Llama-3-8B-Instruct model.

We analyzed the biases of GPT-4 and the fine-tuned Llama 3 model during verification (Figure 3). We found that GPT-4 exhibited a negative bias, such as a higher occurrence of *False* judgments and a tendency to classify *Half* claims as *False*, reflecting GPT-4's cautious bias, which influenced the model's performance. In contrast, the fine-tuned Llama 3 model displayed a more balanced bias, particularly improving accuracy when judging *Half* and *True* statements. These observations suggest that through fine-tuning, the Llama 3 model can better adapt to multidimensional verification tasks when handling complex statements, thereby enhancing the stability and reliability of the inference process.

5.2 Ablation Study

An ablation study was conducted to assess the impact of different configurations on EVICheck, as shown in Figure 4.

Effect of Model Fine-Tuning. We first evaluated the effect of fine-tuning on EVICheck. Fine-tuning the Llama 3 model using X_{train} improved final prediction accuracy by 3.47% compared to the untuned model, highlighting fine-tuning's key role in enhancing performance.

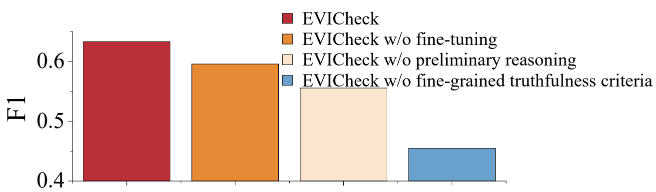

Figure 4: Experimental results of ablation study.

Effect of Preliminary Reasoning. In the original setup, the model performed preliminary reasoning—generating background, extracting evidence, analyzing causal relationships, and concluding—before combining and validating the results at the final verification node. Without preliminary reasoning, the model only gathered evidence, leading to a 7.77% performance drop. This performance drop highlights the preliminary reasoning's key role in integrating multi-round inference and improving accuracy.

Effect of Fine-Grained Truthfulness Criteria. We tested the influence of fine-grained truthfulness criteria. Removing these, along with preliminary reasoning, led to poor performance. However, incorporating our fine-grained prompt words boosted EVICheck performance by 10.08%, showing that detailed criteria significantly enhance the model's ability to perform accurate verification.

5.3 Optimal Solution for the Number of Loop Rounds and Verification Questions

To explore the optimal solution for the number of loops and verification questions, we randomly selected 12 samples (4 from each category). A total score of 12 points was assigned, with 1 point awarded for correct answers, 0 points for incorrect answers, and a 0.3-point deduction for significant errors (such as *true-false* or *false-true* discrepancies). The final score rate was computed as the ratio of the obtained score to the total score.

Number of Loop Rounds. As shown in Figure 5 (left), the number of loops was increased while fixing 2 validation questions per round. It was observed that the score rate increased with the number of loops. However, each additional loop also reduced the inference speed by 50% and increased the API call error rate. Therefore, considering both performance and efficiency, we set the number of loops to 2.

Number of Verification Questions. As shown in Figure 5 (right), the impact of varying the number of validation questions on model accuracy. With the number of loops fixed at 2, an inverted U-shaped curve in accuracy was observed.

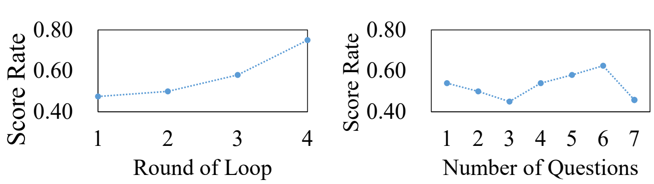

Figure 5: Left: Performance with loop rounds. Right: Performance with the number of verification questions.

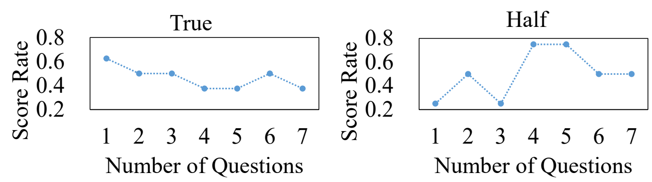

Figure 6: Left: Performance of data labeled as True. Right: Performance of data labeled as Half.

Specifically, the score rate was highest when 6 questions were posed. However, as the number of questions increased, the score rate decreased after reaching its peak. Notably, when only 1 question was asked, the score remained relatively high. We conducted a deeper analysis. Specifically, we analyzed performance for Half, True, and False declarations across different question numbers:

a) True: As shown in Figure 6 (left), the score for *True* decreased as the number of questions increased. With only 1 question, queries were broad, making counterevidence harder to find, leading to more *True* classifications. With more questions, the queries became more specific, reducing the number of *True* classifications.

b) Half: As shown in Figure 6 (right), the score for *Half* statements followed by an inverted U-shape. More questions made the queries more specific, helping the model assess complex statements better. However, too many questions introduced noise, reducing accuracy.

c) False: Accuracy for *False* statements remained at 100%, demonstrating the model's robustness in identifying false statements.

Additionally, fluctuations in experimental results may be influenced by data size, and future studies with larger sample sizes could further validate these conclusions.

5.4 Human Evaluation

To evaluate the performance of EVICheck in generating judgments and explanations, three experts in NLP and public opinion analysis manually reviewed the results. Each expert rated the judgments and explanations on a scale from 1 (poor) to 5 (good) based on predefined criteria.

Twelve data points were randomly selected, and two types of explanations were compared: those from the RAWFC dataset and those generated by EVICheck. Experts rated both based on the following criteria:

- **Coverage:** Whether the explanation covers the key information needed for judgment verification.
- **Readability:** Whether the language of the explanation is concise and well-structured.
- **Accuracy:** Whether the explanation accurately reflects the data or facts and whether the reasoning is correct.
- **Conciseness:** Whether the explanation is succinct and contains only necessary information.
- **Credibility:** Whether the explanation is reasonable and convincing.

	RAWFC	EVICheck
Coverage	3.41	**4.16**
Readability	3.82	**4.34**
Accuracy	4.25	**4.30**
Conciseness	**4.11**	3.56
Credibility	3.87	**4.23**

Table 4: Evaluation results of RAWFC and EVICheck methods across different criteria.

To reduce bias, we randomized the order of the explanations in each questionnaire. Expert ratings were then summarized, and the comparison of scores for each data point across criteria is shown in Table 4.

The results showed that EVICheck received a lower score in *Conciseness*, as manual explanations tend to be more concise. EVICheck includes reasoning steps for each loop, providing transparency but adding unnecessary details.

5.5 Case Study

This case study demonstrates our method's application in verification tasks. The target claim is shown in Figure 7.

Figure 7: The claim to be verified.

The First-Round Verification Questions. Figure 8 shows five generated questions, with the most relevant selected for web retrieval. GPT-4 then performs preliminary inference using the retrieved information.

The Second-Round Verification Questions. As shown in Figure 9, five questions are posed based on the claim and the optimal question from the first round. The most relevant question is selected for web retrieval, and GPT-4 provides further inferences and answers.

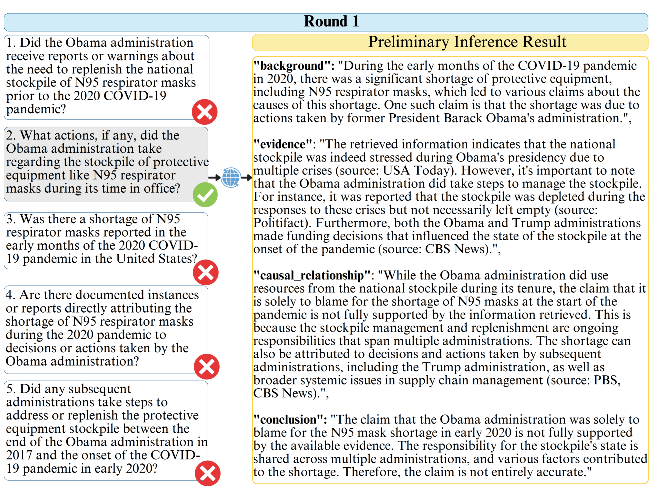

Figure 8: First-round verification questions and initial reasoning.

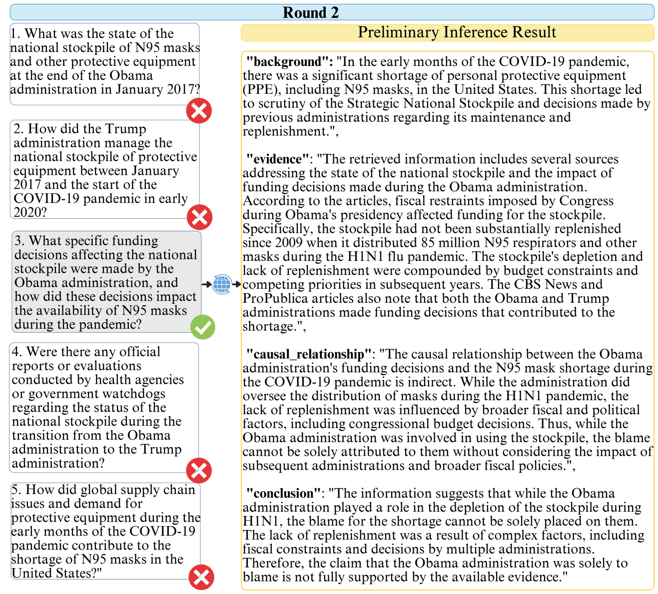

Figure 9: Second-round iterative questioning and refined reasoning.

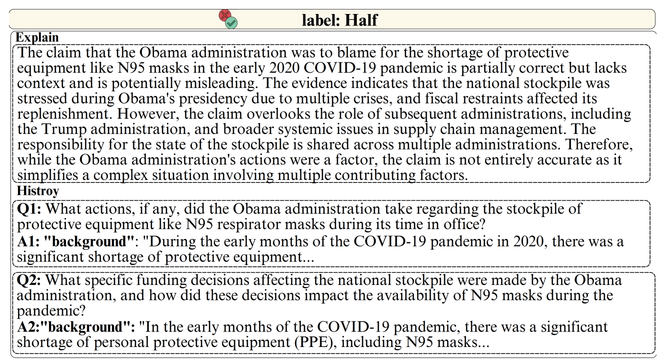

Figure 10: Final judgment based on combined verification and fine-grained truthfulness criteria.

Combined Verification and Final Judgment. Figure 10 compiles the preliminary inferences from both rounds and their evidence, then applies combined verification to make the final judgment based on fine-grained truthfulness criteria.

6 Conclusion

In this paper, we propose EVICheck, a method that enhances automated fact-checking by addressing the limitations of insufficient evidence utilization and the lack of clear verification standards. EVICheck analyzes each piece of evidence independently, conducts detailed reasoning, integrates the results, and applies fine-grained truthfulness criteria to improve reliability. Experiments on the RAWFC dataset show that EVICheck outperforms existing approaches, demonstrating its potential. Nevertheless, it still struggles with informal social-media statements and multimodal claims. Future work will integrate additional social-media APIs and strengthen multimodal reasoning. In conclusion, EVICheck offers an innovative, practical solution for combating fake news.

Acknowledgments

This work was supported by the Joint Fund Key Program of the National Natural Science Foundation of China (No. U23B2029), by the Beijing Natural Science Foundation (No. 4252035), by the Key R&D Program of Zhejiang (No. 2024C01164), by the Zhejiang Provincial Natural Science Foundation of China (No. LMS25F020005), by the National Natural Science Foundation of China (No. 62002027), by the Fundamental Research Funds for the Central Universities (No. CUC25SG013), and by the Public Computing Cloud, CUC.

References

[Atanasova *et al.*, 2020] Pepa Atanasova, Jakob Grue Simonsen, Christina Lioma, and Isabelle Augenstein. Generating fact checking explanations. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7352–7364, Online, 2020. Association for Computational Linguistics.

[Borgeaud *et al.*, 2022] Sebastian Borgeaud, Arthur Mensch, Jordan Hoffmann, Trevor Cai, Eliza Rutherford, Katie Millican, George Bm Van Den Driessche, Jean-Baptiste Lespiau, Bogdan Damoc, Aidan Clark, et al. Improving language models by retrieving from trillions of tokens. In *International Conference on Machine Learning*, pages 2206–2240. PMLR, 2022.

[Brown *et al.*, 2020] Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel Ziegler, Jeffrey Wu, Clemens Winter, Chris Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. Language models are few-shot learners. In *Advances in Neural Information Processing Systems*, volume 33, pages 1877–1901. Curran Associates, Inc., 2020.

[Chen *et al.*, 2024] Guangyao Chen, Siwei Dong, Yu Shu, Ge Zhang, Jaward Sesay, Börje Karlsson, Jie Fu, and Yemin Shi. AutoAgents: a framework for automatic agent generation. In *Proceedings of the Thirty-Third International Joint Conference on Artificial Intelligence*, pages 22–30. International Joint Conferences on Artificial Intelligence Organization, 2024.

[Chern *et al.*, 2023] I-Chun Chern, Steffi Chern, Shiqi Chen, Weizhe Yuan, Kehua Feng, Chunting Zhou, Junxian He, Graham Neubig, Pengfei Liu, et al. FacTool: factuality detection in generative AI–a tool augmented framework for multi-task and multi-domain scenarios. *arXiv preprint arXiv:2307.13528*, 2023.

[Guu *et al.*, 2020] Kelvin Guu, Kenton Lee, Zora Tung, Panupong Pasupat, and Mingwei Chang. Retrieval augmented language model pre-training. In *International Conference on Machine Learning*, pages 3929–3938. PMLR, 2020.

[Hu *et al.*, 2022] Edward J Hu, yelong shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. LoRA: Low-rank adaptation of large language models. In *International Conference on Learning Representations*, 2022.

[Izacard and Grave, 2021] Gautier Izacard and Edouard Grave. Leveraging passage retrieval with generative models for open domain question answering. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics*, pages 874–880, Online, 2021. Association for Computational Linguistics.

[Khaliq *et al.*, 2024] Mohammed Abdul Khaliq, Paul Yu-Chun Chang, Mingyang Ma, Bernhard Pflugfelder, and Filip Miletić. RAGAR, your falsehood radar: RAG-augmented reasoning for political fact-checking using multimodal large language models. In *Proceedings of the Seventh Fact Extraction and VERification Workshop (FEVER)*, pages 280–296, Miami, Florida, USA, 2024. Association for Computational Linguistics.

[Kotonya and Toni, 2020] Neema Kotonya and Francesca Toni. Explainable automated fact-checking for public health claims. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing*, pages 7740–7754, Online, 2020. Association for Computational Linguistics.

[Lewis *et al.*, 2020] Patrick Lewis, Ethan Perez, Aleksandra Piktus, Fabio Petroni, Vladimir Karpukhin, Naman Goyal, Heinrich Küttler, Mike Lewis, Wen-tau Yih, Tim Rocktäschel, Sebastian Riedel, and Douwe Kiela. Retrieval-augmented generation for knowledge-intensive NLP tasks. In *Advances in Neural Information Processing Systems*, volume 33, pages 9459–9474. Curran Associates, Inc., 2020.

[Ma *et al.*, 2019] Jing Ma, Wei Gao, Shafiq Joty, and Kam-Fai Wong. Sentence-level evidence embedding for claim verification with hierarchical attention networks. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 2561–2571, Florence, Italy, 2019. Association for Computational Linguistics.

[Ostrowski *et al.*, 2021] Wojciech Ostrowski, Arnav Arora, Pepa Atanasova, and Isabelle Augenstein. Multi-hop fact checking of political claims. In *Proceedings of the Thirtieth International Joint Conference on Artificial Intelligence*, pages 3892–3898. International Joint Conferences on Artificial Intelligence Organization, 2021.

[Peng *et al.*, 2023] Baolin Peng, Michel Galley, Pengcheng He, Hao Cheng, Yujia Xie, Yu Hu, Qiuyuan Huang, Lars Liden, Zhou Yu, Weizhu Chen, et al. Check your facts and try again: improving large language models with external knowledge and automated feedback. *arXiv preprint arXiv:2302.12813*, 2023.

[Shang *et al.*, 2022] Lanyu Shang, Yang Zhang, Zhenrui Yue, YeonJung Choi, Huimin Zeng, and Dong Wang. A knowledge-driven domain adaptive approach to early misinformation detection in an emergent health domain on social media. In *IEEE/ACM International Conference on*

Advances in Social Networks Analysis and Mining, pages 34–41. IEEE, 2022.

[Thorne *et al.*, 2018] James Thorne, Andreas Vlachos, Christos Christodoulopoulos, and Arpit Mittal. FEVER: a large-scale dataset for Fact Extraction and VERification. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 809–819, New Orleans, Louisiana, 2018. Association for Computational Linguistics.

[Wang and Shu, 2023] Haoran Wang and Kai Shu. Explainable claim verification via knowledge-grounded reasoning with large language models. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 6288–6304, Singapore, 2023. Association for Computational Linguistics.

[Wang *et al.*, 2023] Boxin Wang, Wei Ping, Peng Xu, Lawrence McAfee, Zihan Liu, Mohammad Shoeybi, Yi Dong, Oleksii Kuchaiev, Bo Li, Chaowei Xiao, Anima Anandkumar, and Bryan Catanzaro. Shall we pretrain autoregressive language models with retrieval? A comprehensive study. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 7763–7786, Singapore, 2023. Association for Computational Linguistics.

[Wei *et al.*, 2022] Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, brian ichter, Fei Xia, Ed Chi, Quoc V Le, and Denny Zhou. Chain-of-thought prompting elicits reasoning in large language models. In *Advances in Neural Information Processing Systems*, volume 35, pages 24824–24837. Curran Associates, Inc., 2022.

[Wu *et al.*, 2022] Junfei Wu, Qiang Liu, Weizhi Xu, and Shu Wu. Bias mitigation for evidence-aware fake news detection by causal intervention. In *Proceedings of the 45th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 2308–2313, New York, NY, USA, 2022. Association for Computing Machinery.

[Yang *et al.*, 2022] Zhiwei Yang, Jing Ma, Hechang Chen, Hongzhan Lin, Ziyang Luo, and Yi Chang. A coarse-to-fine cascaded evidence-distillation neural network for explainable fake news detection. In *Proceedings of the 29th International Conference on Computational Linguistics*, pages 2608–2621, Gyeongju, Republic of Korea, 2022. International Committee on Computational Linguistics.

[Yue *et al.*, 2024] Zhenrui Yue, Huimin Zeng, Lanyu Shang, Yifan Liu, Yang Zhang, and Dong Wang. Retrieval augmented fact verification by synthesizing contrastive arguments. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics*, volume 1, pages 10331–10343, Bangkok, Thailand, 2024. Association for Computational Linguistics.

[Zamani and Bendersky, 2024] Hamed Zamani and Michael Bendersky. Stochastic RAG: end-to-end retrieval-augmented generation through expected utility maximization. In *Proceedings of the 47th International ACM SIGIR Conference on Research and Development in Information Retrieval*, page 2641–2646, New York, NY, USA, 2024. Association for Computing Machinery.

[Zhang and Gao, 2023] Xuan Zhang and Wei Gao. Towards LLM-based fact verification on news claims with a hierarchical step-by-step prompting method. In *Proceedings of the 13th International Joint Conference on Natural Language Processing and the 3rd Conference of the Asia-Pacific Chapter of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 996–1011, Nusa Dua, Bali, 2023. Association for Computational Linguistics.

[Zhao *et al.*, 2023] Runcong Zhao, Miguel Arana-catania, Lixing Zhu, Elena Kochkina, Lin Gui, Arkaitz Zubiaga, Rob Procter, Maria Liakata, and Yulan He. PANACEA: An automated misinformation detection system on COVID-19. In *Proceedings of the 17th Conference of the European Chapter of the Association for Computational Linguistics: System Demonstrations*, pages 67–74, Dubrovnik, Croatia, 2023. Association for Computational Linguistics.

[Zheng *et al.*, 2024] Yaowei Zheng, Richong Zhang, Junhao Zhang, Yanhan Ye, and Zheyan Luo. LlamaFactory: unified efficient fine-tuning of 100+ language models. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics*, volume 3, pages 400–410, Bangkok, Thailand, 2024. Association for Computational Linguistics.