Summary Factual Inconsistency Detection Based on LLMs Enhanced by Universal Information Extraction

Anguo Li
Beihang University
Sino-French Engineer School
anguoli@buaa.edu.cn

Lei Yu
Beihang University
Sino-French Engineer School
yulei@buaa.edu.cn

Abstract

Automatic text summarization has a potential flaw that affects the factuality of summaries. Recently, Large Language Models (LLMs) have been introduced as detectors for factual inconsistencies in summaries. However, LLM-based methods rely on reasoning capabilities and face challenges in terms of efficiency and explainability. We focus on decoupling LLMs' information extraction and reasoning capabilities to address prominent challenges, and propose a novel framework, UIEFID (Universal Information Extraction-enhanced Factual Inconsistency Detection). Our idea is to define a self-adaptive structured schema to guide fine-tuned LLMs in extracting unified structured information from documents and summaries, ultimately detecting the origins of inconsistencies in extraction information. The evaluation on 5 open-source models shows that UIEFID not only enhances the detection accuracy on the AGGREFACT benchmark but also significantly reduces redundant reasoning.

1 Introduction

Automatic text summarization compresses extensive source documents into brief summaries, covering diverse content like news articles, source codes, and cross-lingual text (Pu et al., 2023). Previous summarization models are categorized into two main paradigms (Varab and Xu, 2023). Extractive models select and combine significant sentences or passages, ensuring faithfulness but potentially lacking coherence and conciseness. Abstractive models generate novel content, producing more natural and fluent summaries but risking faithfulness issues. Recently, Large language models (LLMs) have demonstrated remarkable capability and open new possibilities for enhancing abstractive summarization (Dhaini et al., 2024; Jin et al., 2024). Nevertheless, LLMs cannot fully guarantee the faithfulness of generated summaries.

In response to factual inconsistency in abstractive summarization, leveraging LLMs as detectors is gaining increasing popularity (Shen et al., 2023; Wang et al., 2023b). However, two main challenges exist in LLM-based methods: **(1) Efficiency.** Current methods heavily rely on the reasoning capability of LLMs. This implies that whether these inconsistencies are explicit or implicit, LLMs typically use extensive reasoning to analyze the discrepancies between summaries and source documents (Luo et al., 2023). Especially for long documents, LLMs struggle to deliver detection results concisely. **(2) Explainability.** LLM-based methods may contain reasoning errors and produce unstable outputs, which can be referred to as "hallucinations" (Zhang et al., 2023). Such inherent flaws render the detection results unreliable and deficient in explainability.

Recent research (Wang et al., 2024; Rettenberger et al., 2024) has demonstrated that LLMs excel in processing structured information. We focus on decoupling LLMs' information extraction and reasoning capabilities to address prominent challenges. By introducing Universal information extraction (UIE) (Lu et al., 2022), we facilitate LLMs to identify patterns, extract key information, and transform unstructured textual data into structured data for more refined detection.

Consequently, we propose a novel detection framework, UIEFID (Universal Information Extraction-enhanced Factual Inconsistency Detection). Our framework obtains a fine-tuned LLM with excellent information extraction capability, enabling it to extract corresponding structured information from both summaries and documents efficiently. By analyzing the discrepancies within structured information, LLM can pinpoint and annotate the origins of inconsistencies, subsequently feeding them back into further refined reasoning to retrieve substantiating evidence from the document to validate inconsistencies.

Findings of the Association for Computational Linguistics: ACL 2025, pages 25450–25465
July 27 - August 1, 2025 ©2025 Association for Computational Linguistics

Our work emphasizes the information extraction capabilities of LLMs rather than relying solely on their zero-shot or few-shot reasoning capabilities. By leveraging UIE, our approach can effectively extract structured information that facilitates easy comparison and verification while directing the trajectory and scope of reasoning. This enhances both the efficiency and explainability of the detection. Our contributions are threefold:

- Our study pioneers the integration of UIE to enhance the efficiency and explainability of LLM-based factual inconsistency detection.
- We propose UIEFID, a novel detetion framework that can rapidly perform and output structured detection results with strong explainability.
- The comprehensive experiments on the popular AGGREFACT benchmark demonstrate the effectiveness of UIEFID, and the quantitative analysis regarding the reduction of redundant reasoning is provided.

2 Approach

2.1 Problem Definition

Given a non-empty document $D = \{d_i\}_{1 \le i \le N}$ and a corresponding summary $S = \{s_j\}_{1 \le j \le M}$, where D contains N sentences and S contains M sentences, typically $N \ge M$, and the text length of S does not exceed that of D. This indicates that converting D into S involves a mapping from multiple sentences to a single sentence, or a mapping from a single sentence to a single sentence.

$$f(\{d_p, \cdots, d_q\}_{1 \le p \le q \le N}) \rightarrow \{s_j\}_{1 \le j \le M} \quad (1)$$

Inspired by text compression algorithms (Li et al., 2022) , we present a concise hypothesis: applying a lossless structured information extractor (IE) to D and S, transforming the semantics and logic of the textual expression into Unified structured information (USI). This hypothesis can be formalized and defined as follows:

$$IE(D, S) \rightarrow (I_d, I_s) \quad (2)$$

where $IE(\cdot)$ is a function that converts document D and summary S into their corresponding structured information, I_d and I_s, respectively.

The information extraction not only completely reflects the semantic entities, logical relationships, and positional markers in text but also significantly reduces the substantial redundancy of content that needs to be processed, accelerating machine detection. As shown in Figure 1, the summary is factually consistent with the document, and the document's structured information should logically entail the summary's structured information.

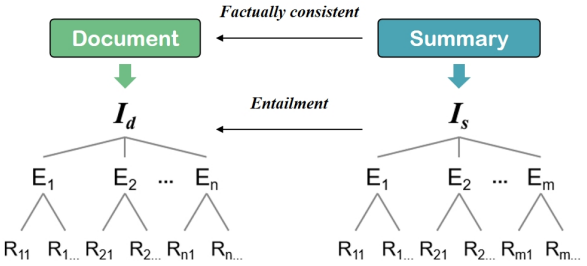

Figure 1: Lossless text compression (tree structure). E_* represents an entity, and R_{**} represents the relationship corresponding to each entity.

In semantics, the subject refers to the entity that performs the action or is described. To express structured information concretely, I_d and I_s can be transformed into a subject-centered triplet format (clarity in structure and ease of processing), described as follows:

$$I_d, I_s \rightarrow \{Sub_i : (k_j, v_j)_{1 \le j \le p_i}\}_{1 \le i \le n/m} \quad (3)$$

where Sub_i denotes the i-th subject in D or S, k_j, v_j respectively denote the j-th key-value pair belonging to Sub_i, which express specific attributes or relations, n and m respectively represent the total number of subjects present in D and S, and p_i indicates the number of key-value pairs possessed by the i-th subject.

Our fundamental idea involves assessing the factuality in the summary by detecting the inconsistency between I_d and I_s.

2.2 UIEFID

To concretize the abovementioned detection idea, we propose a novel detection framework **UIEFID.** It adopts a sequential strategy that combines detection and revision for inconsistency, divided into three phases: **Subject Alignment**, **Key-value Analysis** and **Factuality Evaluation**.

2.2.1 Subject Alignment

A summary typically consists of statements involving semantic elements such as subject, action, object, modification, and state assertions. In this paper, the term *"subject"* is used to encompass all

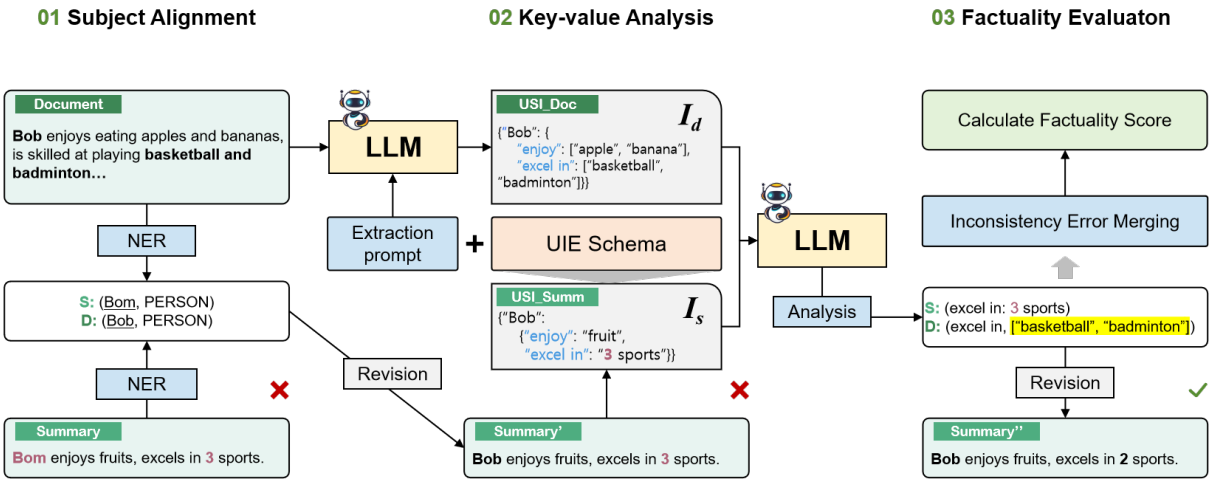

Figure 2: UIEFID framework consists of 3 phases: **(1) Subject Alignment.** Identifying and correcting inconsistencies in subjects between the document and the summary. **(2) Key-value Analysis.** Extracting USI from the summary and using it as a extraction schema to guide the fine-tuning LLM to extract USI from the document, and then prompting LLM to reason and analyze the differences in structured key-value pairs and obtain the detection results. **(3) Factuality Evaluation.** Collecting inconsistencies from two phases and merging them to calculate the final factuality score.

terms related to named entities, which is equivalent to the intermediate nodes of a tree-structured information (see Figure 2).

In this phase, we compare the subject differences in the structured information by Named entity recognition (NER) , and identify three types of errors: *non-existent*, *spelling errors*, and *misplacement*. Specifically, ***non-existent*** indicates that the subject in the summary does not have a corresponding subject in the source document; ***spelling errors*** means that the subject in the summary was not correctly output; ***misplacement*** refers to the relative position of the subject in the summary being different from that in the source document.

After identifying inconsistencies, we perform text similarity computation and coreference resolution to select appropriate subjects from the document to replace the inconsistent content in the summary. Meanwhile, the analysis results of subject alignment are recorded as a basis for subsequent calculations of the factuality score.

Clearly, subject misalignment represents a fundamental semantic error. Significant discrepancies between subjects extracted from the summary and the document can mislead LLMs into over-reasoning and introduce biases during detection. Thus, it is imperative to promptly correct subject errors to ensure that LLMs can focus on more complex logical inconsistencies.

2.2.2 Key-value Analysis

After eliminating subject misalignment, we fine-tune LLMs specifically for information extraction tasks to extract structured information (subject: key-value pairs) from summaries, akin to constructing triples for knowledge graphs. We then mask these values to form extraction schemas, which guide the LLMs to retrieve relevant content from the document and fill in the masked values (for examples, see Appendix A). Finally, we directly compare the differences between the key-value pairs extracted from the structured information in the summary and the document to determine the source of inconsistencies.

Existing LLM-based methods typically rely on the document as a reference for detecting inconsistencies within summaries (Shen et al., 2023; Wang et al., 2023b). However, a primary drawback of this approach is its high sensitivity to document length, and longer documents can significantly degrade performance due to the increased complexity and computational load associated with inconsistency detection. Our approach redefines the focus of inconsistency detection. Given that summaries are typically much more concise than their source documents, we base our verification on the summary text to retroactively validate whether there exists any structured information in the document that contradicts the summary.

2.2.3 Factuality Evalution

To enhance the explainability, we quantify the accumulated consistencies after detection and revision, thereby computing the factuality score (FS) of the summary. The computation formula is as follows:

$$FS = \frac{|\text{ents}^{fc}|}{|\text{ents}|} \times \frac{\sum_{i=1}^{|\text{Sub}|} \frac{|p_i^{fc}|}{|p_i|}}{|\text{Sub}|} \quad (4)$$

where ents^{fc} and p_i^{fc} represent factually consistent entities in each summary and key-value pairs for i-th subject, $|\cdot|$ represents the corresponding quantity. This computation ensures that high factuality scores are achieved only when both entities and key-value pairs exhibit a high degree of consistency with the document.

The factuality score serves as a metric to evaluate the degree of consistency between the summary and the document. By leveraging the document as a reference, we compute the proportion of entities in the summary that align with those in the document. Additionally, we evaluate the proportion of key-value pair matches within the structured information extracted through the universal information extraction. This dual approach provides a comprehensive evaluation of summary factuality.

3 Experiment and Results

3.1 Experimental settings

We introduce experimental settings, including the benchmark, evaluation metric, and baselines.

Benchmark. We evaluate the effectiveness of UIEFID on AGGREFACT, the established benchmark for assessing summarization factuality metrics. AGGREFACT aggregates 9 existing annotated summary factuality datasets of news articles. All datasets contain summaries generated from articles in CNN/DM and XSum. Given the unique characteristics of CNN/DM and XSum, our proposed benchmark includes two subsets, AGGREFACT-CNN and AGGREFACT-XSUM, that evaluate the performance of factuality metrics on these two datasets separately. Tang et al., 2023 stratify it according to the underlying summarization model, categorized into FTSOTA, EXFORMER and OLD based on their development timeline. (1) FTSOTA represents state-of-the-art fine-tuned summarization models, including BART (Lewis et al., 2020), PEGASUS (Zhang et al., 2020), and T5 (Raffel et al., 2020). (2) EXFORMER is a collection of early Transformer-based summarization models, comprising BERTSum (Liu and Lapata, 2019) and GPT-2 (Radford et al., 2019). (3) OLD covers the remaining models, such as Pointer-Generator (See et al., 2017) and BottomUp (Gehrmann et al., 2018).

Evaluation Metric. We use balanced accuracy to evaluate the performance of factual inconsistency detection methods due to the imbalance of factually consistent and inconsistent summaries in AGGREFACT dataset.

Baselines. We evaluate the effectiveness of UIEFID against 12 recent baselines. They are categorized into three types based on their detection pattern: NLI, QA, and LLM-based metrics (see section 4) .

3.2 Implementation Details

Filtering and Alignment. Employing LLMs directly to detect elementary errors such as misaligned subjects is unnecessary. Instead, we utilize the low-resource NLP tool *spaCy's en-core-web-trf* (Jugran et al., 2021) to extract inconsistent subjects in document-summary pairs. We then leverage its built-in similarity computation function to re-match the subjects in the document that align with those in the summary, thereby achieving preliminary subject alignment.

Fine-tuning. High-quality instruction data is the vital key for enhancing the specific capabilities of LLMs. We first fine-tuned the base models using the open-source IEPile (Gui et al., 2024) (a comprehensive bilingual [English and Chinese] information extraction instruction corpus containing approximately 0.32B tokens) to enhance their performance in information extraction, including the three primary tasks of named entity recognition (NER), relation extraction (RE), and event extraction (EE) (Niklaus et al., 2018). We selected Llama 3-8B (Meta, 2024), Llama 3.1-8B, Qwen2.5 (7B, 14B) (Qwen, 2025) for fine-tuning to facilitate subsequent performance comparison and result analysis. Additionally, we invoked the advanced DeepSeek-R1 (DeepSeek-AI, 2025) API to test the improvement of our framework's detection performance.

Schema Design. Inspired by the Structured extraction language for UIE (Lu et al., 2022), we designed a schema for LLMs to perform structured extraction after fine-tuning (see Appendix B). For information extraction, the schema is typically constructed in a JSON-like format to ensure that

the responses are uniformly formatted and readable

Detection and Revision. LLMs are instructed to extract structured information from the summary and mask the detailed values of key-value pairs, retaining only the subjects and corresponding keys. We use the masked format as the schema to guide LLMs in filling in new values according to the document. We can quickly detect inconsistencies by comparing the differences between the structured information extracted from the document and the summary. We directly prompt LLMs to revise and polish the summary by addressing the identified sources of factual inconsistencies. (prompt templates, see Appendix C)

3.3 Results and Analysis

Table 1: Balanced acccuracy results on the test sets of the AGGREFACT-CNN and AGGREFACT-XSUM datasets (threshold-per-dataset setting).

	Agg-CNN			Agg-XSum			
	FTS	**ExF**	**OLD**	**FTS**	**ExF**	**OLD**	**AVG**
Baseline	50.0	50.0	50.0	50.0	50.0	50.0	50.0
DAE*	59.4	67.9	69.7	73.1	-	-	67.5
QuestEval	63.7	64.3	65.2	61.6	60.1	59.7	63.7
SummaC-ZS	63.3	76.5	76.3	56.1	51.4	53.3	68.1
SummaC-Conv	70.3	69.8	78.9	67.0	64.6	67.5	71.5
TrueTeacher-11B	62.0	67.5	80.5	75.9	68.4	52.8	71.4
QAFactEval	61.6	69.1	80.3	65.9	59.6	60.5	69.2
MENLI	63.4	54.9	66.8	59.0	59.7	70.5	61.0
AlignScore	62.7	73.2	78.0	69.4	77.5	63.7	70.8
ChatGPT-ZS	66.2	64.5	74.3	62.6	69.2	60.1	66.9
ChatGPT-CoT	49.7	60.4	66.7	56.0	60.9	50.1	58.2
ChatGPT-DA	48.0	63.6	71.0	53.6	65.6	61.5	59.1
ChatGPT-Star	55.8	65.8	71.2	57.7	70.6	53.8	62.6
Llama3-8B	61.3	59.7	63.4	60.2	69.2	59.3	61.2
+*fine-tuning*	63.1	62.9	64.2	62.4	72.5	61.3	63.2
+*UIEFID*	68.6	76.8	77.2	71.3	76.6	70.2	73.5
Llama3.1-8B	63.5	61.8	67.4	63.2	71.3	63.2	64.0
+*fine-tuning*	64.2	63.9	69.5	65.3	72.5	66.8	65.7
+*UIEFID*	75.6	78.4	81.2	73.8	81.3	72.4	77.3
Qwen2.5-7B	56.7	59.3	61.7	60.2	66.3	59.7	59.5
+*fine-tuning*	58.2	60.4	62.2	63.4	68.5	60.9	61.1
+*UIEFID*	65.8	71.2	75.6	72.5	78.3	67.3	71.3
Qwen2.5-14B	65.7	64.2	69.1	61.6	71.2	64.7	65.2
+*fine-tuning*	66.4	66.8	71.3	57.8	72.8	66.3	65.6
+*UIEFID*	78.9	80.7	83.4	76.1	80.5	78.7	79.8
DeepSeek-R1	75.8	82.3	85.7	77.5	84.2	86.4	80.3
+*UIEFID*	**80.4**	**85.3**	**87.4**	**81.6**	**90.1**	**89.5**	**83.7**

We evaluate open-source LLMs under three distinct strategies: zero-shot without fine-tuning, zero-shot with fine-tuning (+*fine-tuning*), and the strategy with fine-tuning and executed in accordance with the UIEFID framework (+*fine-tuning*+*UIEFID*). Due to the limitations of hardware resources, we are unable to deploy and fine-tune DeepSeek-R1 locally, and can only conduct experiments through the API.

3.3.1 Acccuracy

Table 1 presents the balanced accuracy scores on the AGGREFACT test sets. A trivial baseline that predicts all examples as factually (in) consistent reports a balanced accuracy of 50%. Following Tang et al., 2023, we exclude the performance of DAE on the EXFORMER (EXF in table) and OLD datasets in the AGGREFACT-XSUM partition, since it was trained on XSumFaith (Goyal and Durrett, 2021) which is part of those splits. Results in **bold** indicate the best performance, while underlined values represent the second best.

DeepSeek-R1 (+*UIEFID*) sets a new state-of-the-art performance on AGGREFACT, resulting in the highest average balanced accuracy score. DeepSeek-R1 is more inclined towards in-depth thinking and reasoning, and its remarkable performance is not unexpected. This demonstrates that breakthroughs in model reasoning capabilities significantly boost factual inconsistency detection.

On the other four open-source LLMs, the experimental results reveal that the zero-shot detection performance of base models without fine-tuning did not significantly outperform other baselines. Upon specific fine-tuning geared towards information extraction, the models exhibited a marginal enhancement in effectiveness. Furthermore, integrating the fine-tuned models into the UIEFID framework and executing the three stages sequentially led to a marked improvement in performance (with a maximum increase of 14.5%)

Overall, the detection performance aligns with the scaling laws (Kaplan et al., 2020), indicating that more extensive model parameters correlate with higher accuracy.

3.3.2 Efficiency

To assess the efficiency of UIEFID, we construct a utility metric denoted as U, defined as:

$$U = \frac{1}{N} \sum_{i=1}^{N} \left(\frac{T_i^Q}{T_i^R} \right) \quad (5)$$

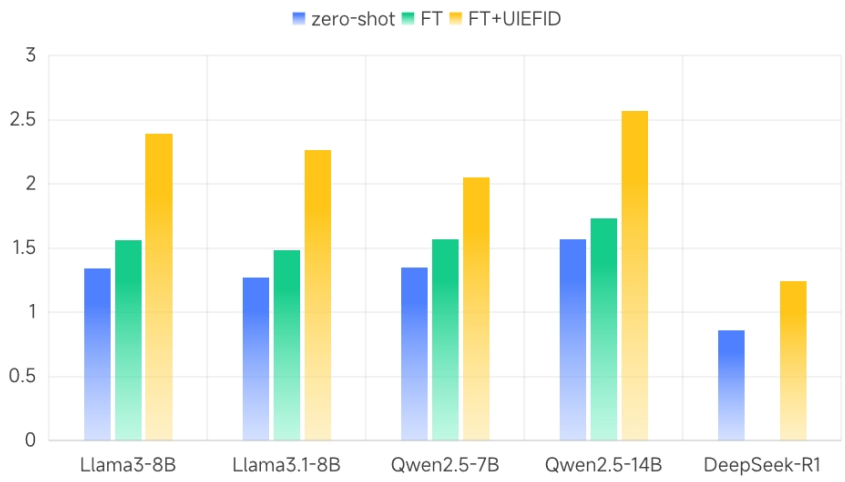

Figure 3: Utility Assessment for Factual Inconsistency Detection (*DeepSeek-R1 skips the fine-tuning step.)

where N represents the number of document-summary pairs. U quantifies the ratio of the number of input tokens (T_i^Q) in the query (encompassing document-summary pairs and prompt) to output tokens (T_i^R) generated by LLMs in response to detected factual inconsistencies.

Figure 3 presents a comparative analysis of the utility metric across different models and strategies for factual inconsistency detection. The models assessed include Llama3-8B, Llama3.1-8B, Qwen2.5-7B, Qwen2.5-14B, and DeepSeek-R1, evaluated under three conditions: zero-shot, fine-tuning (FT), and fine-tuning combined with UIEFID (FT+UIEFID).

The results indicate that the FT+UIEFID strategy consistently outperforms the other two strategies across all models, suggesting that integrating UIEFID significantly enhances the efficiency of factual inconsistency detection. This is evidenced by the higher utility values achieved, which imply a more favourable ratio of input to output tokens, thereby reducing the generation of redundant information. Appendix D further analyzes the significant variations in the proportions of reasoning and extraction within the responses LLMs as document length increases. This indicates that UIEFID can effectively suppress the divergence of reasoning.

However, DeepSeek-R1 exhibits the lowest utility value under the UIEFID strategy, possibly due to its requirement to incorporate additional reasoning and thinking in its responses (DeepSeek-AI, 2025). While this approach enhances the transparency and explainability of the responses, it also significantly increases the number of output tokens. Although DeepSeek-R1 may perform well in accuracy, its efficiency in generating concise and efficient responses may not be as high as that of other models.

3.4 Ablation Study

UIEFID framework treats subject alignment as a preprocessing step for detecting document-summary pairs, with the aim of effectively eliminating subject-related errors. Theoretically, this design prevents LLMs from overcomplicating otherwise simple problems in subsequent analysis, thereby avoiding prolonged reasoning processes that could ultimately lead to instability in key-value analysis outcomes.

To validate the necessity of subject alignment, we conduct an ablation study on fine-tuned Qwen2.5-7B under three configurations: **ZS** (zero-shot detection without UIEFID components), **KVA** (key-value analysis without subject alignment), **SA+KVA** (full UIEFID framework). In the context of ZS configuration, the absence of framework limitations and constraints often leads to uncontrollable detection outcomes. This is manifested by the possibility of obtaining two contradictory judgment results when the same document-summary pair is subjected to multiple detections. Consequently, we repeat the entire experiment ten times and record the final results and range of variation.

As shown in Table 2, the full UIEFID framework (SA+KVA) achieves the highest balanced accuracy on both Agg-CNN (72.6) and Agg-XSum (75.3) subsets, with an average of 73.9. This demonstrates clear performance gains over both ZS (59.3) and KVA (70.2). The results indicate that subject alignment (SA) and key-value analysis (KVA) have com-

Table 2: Balanced accuracy across the three configurations (ΔAVG denotes the average accuracy gain relative to the preceding configuration).

Config	Agg-CNN	Agg-XSum	AVG	ΔAVG
ZS	56.3 ± 0.8	62.4 ± 0.3	59.3	–
KVA	68.2 ± 2.1	72.2 ± 0.2	70.2	+10.9
SA+KVA	$\mathbf{72.6 \pm 2.4}$	$\mathbf{75.3 \pm 0.6}$	**73.9**	+3.7

plementary effects: KVA alone improves average balanced accuracy by 10.9 points over ZS, while SA provides an additional 3.7-point gain. Notably, the moderate increase in range suggests that performance improvements are achieved without introducing instability.

3.5 Robustness

To further assess the robustness of the UIEFID framework, it is important to carefully consider whether the length disparity between the document and the summary affects the performance of inconsistency detection. We partition the AGGREFACT dataset into five clusters based on summary compression ratios (defined as document-to-summary word count ratio): [0, 10×], (10×, 20×], (20×, 30×], (30×, 40×], and (40×, ∞). Subsequently, four fine-tuned LLMs are deployed within our framework to detect factual inconsistencies, and their accuracy is evaluated under varying compression conditions. The corresponding results are illustrated as line plots in Figure 4.

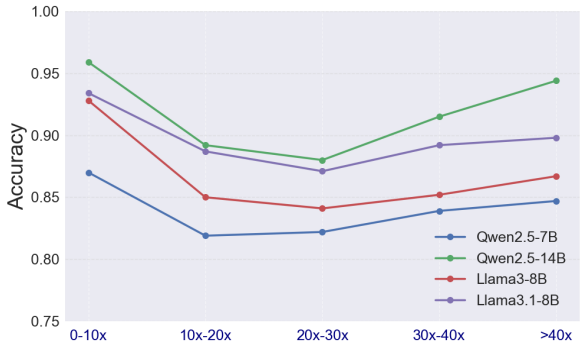

Figure 4: Accuracy across different summary compression ratios.

Empirical results indicate that Qwen2.5-14B outperforms the other three models, which aligns with the common expectation that larger models perform better. However, the relationship between summary compression ratio and model accuracy does not exhibit a strictly linear trend: accuracy initially declines with increasing compression, then gradually recovers. This non-linear behavior is partially attributable to data distribution, and it also suggests that UIEFID maintains relatively stable performance when handling extremely simple or highly compressed document-summary pairs, while exhibiting more variability in intermediate compression ranges.

This finding underscores the necessity of accounting for document-summary length disparities when evaluating LLM-based factuality detection frameworks. The fluctuations in accuracy across different compression levels indicate that input complexity can significantly affect the model's reasoning and prediction accuracy.

4 Related Work

4.1 Factual Inconsistency Detection

Recently, research on factual inconsistency detection (also known as factuality evaluation) in text summarization has been highly active. The current detection metrics mainly fall into the following three paradigms.

NLI-based metrics. The Dependency Arc Entailment (Goyal and Durrett, 2020, DAE) leverage the entailment between the dependency parse trees of document sentences and the summary. SummaC-Cov (Laban et al., 2022) enables NLI models to be successfully used for this task by segmenting documents into sentence units and aggregating scores between pairs of sentences. Chen and Eger, 2023 proposed a series of NLI-based indicators for evaluating the factuality of summaries. They utilized the source document as the premise for the NLI system and the entire summary as the hypothesis for verification. AlignScore (Zha et al., 2023) provides a novel alignment system which aligns the summary with large blocks of text from the source document and provides an overall score for entire summary.

QA-based metrics. This category of metrics typically depend on a Question Generation (QG) model designed to formulate questions based on the input document as context, and check whether information in the summary can be used to answer such questions, and vice versa. FEQA (Durmus et al., 2020), QAFactEval (Fabbri et al., 2022) and QuestEval (Scialom et al., 2021) are representative approachs among these metrics.

LLM-based metrics. Recent studies (Shen et al., 2023; Wang et al., 2023b) have shown that LLM-based metrics exhibit stronger evaluation performance compared to NLI-based metrics. TrueTeacher (Gekhman et al., 2023) analyzed the LLMs' capability of generating large-scale factuality datasets and then trained smaller student models on such data, outperforming existing NLI metrics. The recent trend of methods revolve around prompt engineering strategies, where LLMs are typically instructed with task descriptions, the input documents, and system-generated summaries. Luo et al., 2023 were among the first to utilize LLMs for the detection of factual inconsistencies, inputting the source document in SUMMAC (Laban et al., 2022) along with its summary into GPT-3.5 for evaluation. Fu et al., 2023 demonstrated the capabilities of LLMs in achieving multi-aspect, customized, and training-free evaluation. G-Eval (Liu et al., 2023) experimented with chain-of-thoughts (Wei et al., 2024a, COT) and form-filling instruction paradigms. However, early attempts have shown that LLMs underperform traditional models due to their limited ability to follow instructions and the absence of an effective detection methodology (Tang et al., 2023).

4.2 Universal Information Extraction

Information Extraction (IE) is a crucial domain in natural language processing that converts plain text into structured knowledge (Xu et al., 2023) . There has been a recent surge of interest in generative IE methods (Qi et al., 2023; Parvez et al., 2021) that adopt LLMs to generate structural information. Xu et al., 2023 categorize universal IE frameworks into two formats: natural language (NL-LLMs based) and code language (Code-LLMs based).

Lu et al., 2022 first propose a unified text-to-structure generation framework named UIE, which can universally model different IE tasks, adaptively generate targeted structures, and collaboratively learn general IE abilities from different knowledge sources. InstructUIE (Wang et al., 2023a) is a UIE framework based on instruction tuning in LLMs, which can uniformly model various information extraction tasks and capture the inter-task dependency. ChatIE (Wei et al., 2024b) is a two-stage framework to transform the zero-shot IE task into a multi-turn question-answering problem. Code4UIE (Parvez et al., 2021) is a universal retrieval-augmented code generation framework based on LLMs,which adopts Python classes to define task-specific schemas of various structural knowledge in a universal way.

5 Conclusion

Text summarization models have a potential flaw that causes the generation of summaries with factual inconsistencies. Inspired by structured semantic extraction and lossless text compression, we employ a novel method, innovatively combining LLMs with Universal Information Extraction (UIE), and propose a novel UIEFID framework. It effectively enhances the efficiency and explainability of summary factual inconsistency detection. We conduct comprehensive experiments to demonstrate the effectiveness of UIEFID on the popular AGGREFACT benchmark and provide a detailed quantitative analysis of whether reducing redundant reasoning. Our experiment results indicate that fully leveraging LLMs' remarkable universal information extraction capabilities is a promising path to improve detection performance. UIEFID can effectively extract factual inconsistencies in summaries and incrementally refine them until they are consistent with source documents. Future research will consider adding a "self-reflection" module to LLM-based text summarization. LLMs actively detect and provide feedback to correct the generated results through frameworks based on UIEFID.

Limitations

UIEFID framework relies on the capabilities of base models. Our findings highlight the improvement in efficiency and explainability brought about by leveraging universal information extraction. However, our results do not establish that structured triplet representations are optimal. Alternative representations may better convey the structured semantic content of the text. Furthermore, our analysis is limited by the absence of summary datasets derived from the latest LLMs and a lack of comprehensive examination of the characteristics inherent to LLM-based text summarization. Due to the lack of maintenance and unresolved runtime issues in the official repositories of certain detection baselines, our ability to extend comparative experiments has been partially constrained.

References

Yanran Chen and Steffen Eger. 2023. MENLI: Robust evaluation metrics from natural language inference. *Transactions of the Association for Computational Linguistics*, 11:804–825.

DeepSeek-AI. 2025. Deepseek-r1: Incentivizing reasoning capability in llms via reinforcement learning. *Preprint*, arXiv:2501.12948.

Mahdi Dhaini, Ege Erdogan, Smarth Bakshi, and Gjergji Kasneci. 2024. Explainability meets text summarization: A survey. In *Proceedings of the 17th International Natural Language Generation Conference*, pages 631–645, Tokyo, Japan. Association for Computational Linguistics.

Esin Durmus, He He, and Mona Diab. 2020. FEQA: A question answering evaluation framework for faithfulness assessment in abstractive summarization. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 5055–5070, Online. Association for Computational Linguistics.

Alexander Fabbri, Chien-Sheng Wu, Wenhao Liu, and Caiming Xiong. 2022. Qafacteval: Improved qa-based factual consistency evaluation for summarization. pages 2587–2601.

Jinlan Fu, See-Kiong Ng, Zhengbao Jiang, and Pengfei Liu. 2023. Gptscore: Evaluate as you desire. *Preprint*, arXiv:2302.04166.

Sebastian Gehrmann, Yuntian Deng, and Alexander Rush. 2018. Bottom-up abstractive summarization. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 4098–4109, Brussels, Belgium. Association for Computational Linguistics.

Zorik Gekhman, Jonathan Herzig, Roee Aharoni, Chen Elkind, and Idan Szpektor. 2023. TrueTeacher: Learning factual consistency evaluation with large language models. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 2053–2070, Singapore. Association for Computational Linguistics.

Tanya Goyal and Greg Durrett. 2020. Evaluating factuality in generation with dependency-level entailment. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 3592–3603, Online. Association for Computational Linguistics.

Tanya Goyal and Greg Durrett. 2021. Annotating and modeling fine-grained factuality in summarization. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 1449–1462, Online. Association for Computational Linguistics.

Honghao Gui, Lin Yuan, Hongbin Ye, Ningyu Zhang, Mengshu Sun, Lei Liang, and Huajun Chen. 2024. Iepile: Unearthing large-scale schema-based information extraction corpus. *Preprint*, arXiv:2402.14710.

Hanlei Jin, Yang Zhang, Dan Meng, Jun Wang, and Jinghua Tan. 2024. A comprehensive survey on process-oriented automatic text summarization with exploration of llm-based methods. *ArXiv*, abs/2403.02901.

Swaranjali Jugran, Ashish Kumar, Bhupendra Singh Tyagi, and Vivek Anand. 2021. Extractive automatic text summarization using spacy in python nlp. In *2021 International Conference on Advance Computing and Innovative Technologies in Engineering (ICACITE)*, pages 582–585.

Jared Kaplan, Sam McCandlish, Tom Henighan, Tom B. Brown, Benjamin Chess, Rewon Child, Scott Gray, Alec Radford, Jeffrey Wu, and Dario Amodei. 2020. Scaling laws for neural language models. *Preprint*, arXiv:2001.08361.

Philippe Laban, Tobias Schnabel, Paul N. Bennett, and Marti A. Hearst. 2022. SummaC: Re-visiting NLI-based models for inconsistency detection in summarization. *Transactions of the Association for Computational Linguistics*, 10:163–177.

Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Veselin Stoyanov, and Luke Zettlemoyer. 2020. BART: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7871–7880, Online. Association for Computational Linguistics.

Kefei Li, Yijiang Jia, Yun-Fei Ji, Baodi Xie, W. Zhang, Ping Lu, and Jincai Chen. 2022. Compression strategy of structured text based on prior dictionary for data distribution system. In *Conference on Advanced Algorithms and Signal Image Processing*.

Yang Liu, Dan Iter, Yichong Xu, Shuohang Wang, Ruochen Xu, and Chenguang Zhu. 2023. G-eval: NLG evaluation using gpt-4 with better human alignment. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 2511–2522, Singapore. Association for Computational Linguistics.

Yang Liu and Mirella Lapata. 2019. Text summarization with pretrained encoders. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 3730–3740, Hong Kong, China. Association for Computational Linguistics.

Yaojie Lu, Qing Liu, Dai Dai, Xinyan Xiao, Hongyu Lin, Xianpei Han, Le Sun, and Hua Wu. 2022. Unified structure generation for universal information extraction. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 5755–5772, Dublin, Ireland. Association for Computational Linguistics.

Zheheng Luo, Qianqian Xie, and Sophia Ananiadou. 2023. Chatgpt as a factual inconsistency evaluator for text summarization. *Preprint*, arXiv:2303.15621.

Meta. 2024. The llama 3 herd of models. *Preprint*, arXiv:2407.21783.

Christina Niklaus, Matthias Cetto, André Freitas, and Siegfried Handschuh. 2018. A survey on open information extraction. In *Proceedings of the 27th International Conference on Computational Linguistics*, pages 3866–3878, Santa Fe, New Mexico, USA. Association for Computational Linguistics.

Md Rizwan Parvez, Wasi Ahmad, Saikat Chakraborty, Baishakhi Ray, and Kai-Wei Chang. 2021. Retrieval augmented code generation and summarization. In *Findings of the Association for Computational Linguistics: EMNLP 2021*, pages 2719–2734, Punta Cana, Dominican Republic. Association for Computational Linguistics.

Xiao Pu, Mingqi Gao, and Xiaojun Wan. 2023. Summarization is (almost) dead. *Preprint*, arXiv:2309.09558.

Ji Qi, Chuchun Zhang, Xiaozhi Wang, Kaisheng Zeng, Jifan Yu, Jinxin Liu, Lei Hou, Juanzi Li, and Xu Bin. 2023. Preserving knowledge invariance: Rethinking robustness evaluation of open information extraction. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 5876–5890, Singapore. Association for Computational Linguistics.

Qwen. 2025. Qwen2.5 technical report. *Preprint*, arXiv:2412.15115.

Alec Radford, Jeff Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. 2019. Language models are unsupervised multitask learners.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu. 2020. Exploring the limits of transfer learning with a unified text-to-text transformer. *J. Mach. Learn. Res.*, 21(1).

Luca Rettenberger, Marc F. Münker, Mark Schutera, Christof M. Niemeyer, Kersten S. Rabe, and Markus Reischl. 2024. Using large language models for extracting structured information from scientific texts. *Current Directions in Biomedical Engineering*, 10(4):526–529.

Thomas Scialom, Paul-Alexis Dray, Sylvain Lamprier, Benjamin Piwowarski, Jacopo Staiano, Alex Wang, and Patrick Gallinari. 2021. QuestEval: Summarization asks for fact-based evaluation. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 6594–6604, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Abigail See, Peter J. Liu, and Christopher D. Manning. 2017. Get to the point: Summarization with pointer-generator networks. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1073–1083, Vancouver, Canada. Association for Computational Linguistics.

Chenhui Shen, Liying Cheng, Xuan-Phi Nguyen, Yang You, and Lidong Bing. 2023. Large language models are not yet human-level evaluators for abstractive summarization. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 4215–4233, Singapore. Association for Computational Linguistics.

Liyan Tang, Tanya Goyal, Alex Fabbri, Philippe Laban, Jiacheng Xu, Semih Yavuz, Wojciech Kryscinski, Justin Rousseau, and Greg Durrett. 2023. Understanding factual errors in summarization: Errors, summarizers, datasets, error detectors. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 11626–11644, Toronto, Canada. Association for Computational Linguistics.

Daniel Varab and Yumo Xu. 2023. Abstractive summarizers are excellent extractive summarizers. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 330–339, Toronto, Canada. Association for Computational Linguistics.

Jianing Wang, Qiushi Sun, Xiang Li, and Ming Gao. 2024. Boosting language models reasoning with chain-of-knowledge prompting. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 4958–4981, Bangkok, Thailand. Association for Computational Linguistics.

Xiao Wang, Weikang Zhou, Can Zu, Han Xia, Tianze Chen, Yuansen Zhang, Rui Zheng, Junjie Ye, Qi Zhang, Tao Gui, Jihua Kang, Jingsheng Yang, Siyuan Li, and Chunsai Du. 2023a. Instructuie: Multi-task instruction tuning for unified information extraction. *Preprint*, arXiv:2304.08085.

Yiming Wang, Zhuosheng Zhang, and Rui Wang. 2023b. Element-aware summarization with large language models: Expert-aligned evaluation and chain-of-thought method. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 8640–8665, Toronto, Canada. Association for Computational Linguistics.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Brian Ichter, Fei Xia, Ed H. Chi, Quoc V. Le, and Denny Zhou. 2024a. Chain-of-thought prompting elicits reasoning in large language models. In *Proceedings of the 36th International Conference on Neural Information Processing Systems*, NIPS '22, Red Hook, NY, USA. Curran Associates Inc.

Xiang Wei, Xingyu Cui, Ning Cheng, Xiaobin Wang, Xin Zhang, Shen Huang, Pengjun Xie, Jinan Xu, Yufeng Chen, Meishan Zhang, Yong Jiang, and Wenjuan Han. 2024b. Chatie: Zero-shot information extraction via chatting with chatgpt. *Preprint*, arXiv:2302.10205.

Derong Xu, Wei Chen, Wenjun Peng, Chao Zhang, Tong Xu, Xiangyu Zhao, Xian Wu, Yefeng Zheng, and Enhong Chen. 2023. Large language models for generative information extraction: A survey. *Preprint*, arXiv:2312.17617.

Yuheng Zha, Yichi Yang, Ruichen Li, and Zhiting Hu. 2023. AlignScore: Evaluating factual consistency with a unified alignment function. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 11328–11348, Toronto, Canada. Association for Computational Linguistics.

Jingqing Zhang, Yao Zhao, Mohammad Saleh, and Peter J. Liu. 2020. Pegasus: Pre-training with extracted gap-sentences for abstractive summarization. *Preprint*, arXiv:1912.08777.

Yue Zhang, Yafu Li, Leyang Cui, Deng Cai, Lemao Liu, Tingchen Fu, Xinting Huang, Enbo Zhao, Yu Zhang, Yulong Chen, Longyue Wang, Anh Tuan Luu, Wei Bi, Freda Shi, and Shuming Shi. 2023. Siren's song in the ai ocean: A survey on hallucination in large language models.

A Key-value Analysis Example

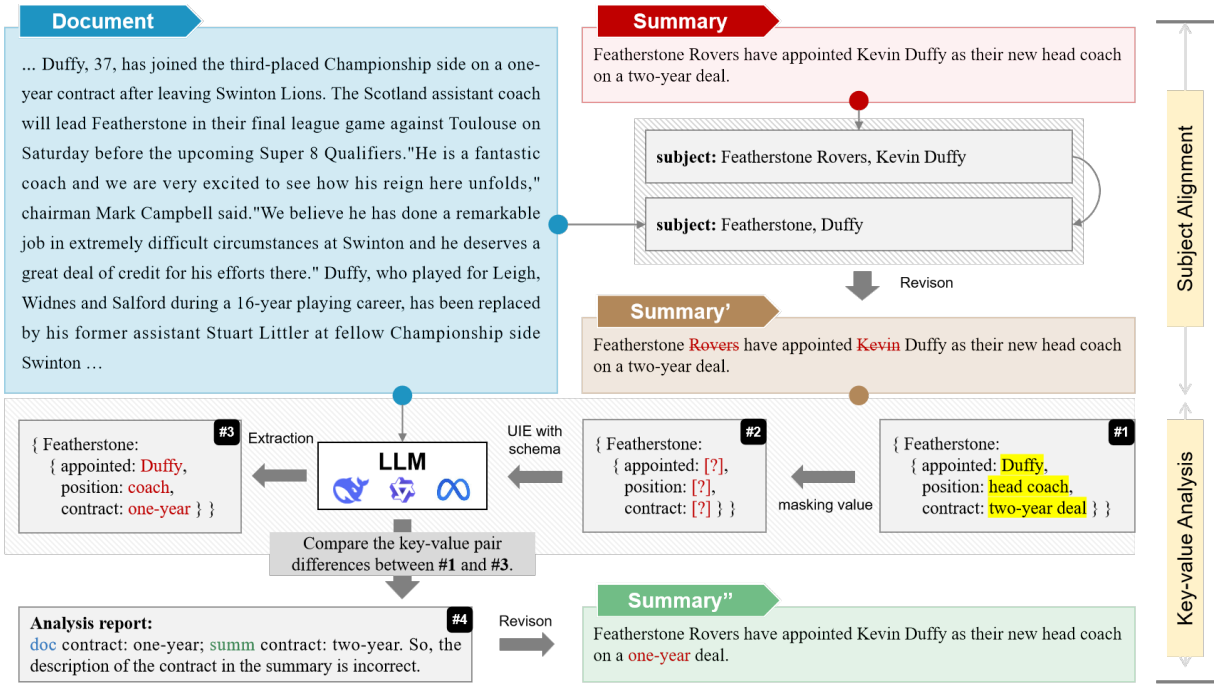

Figure 5: A process of key-value analysis based on an example. It describes a process of key-value analysis based on an example. Subject alignment mitigates discrepancies in subjects between the document and the summary. We observe that the summary supplements personal names, without access to external verification, it is challenging to ascertain the absolute accuracy of these names in the summary. Consequently, we replace them with names that are present in the document. For the key-value analysis, we initially instruct LLMs to extract structured information (#1) from the revised summary. Subsequently, we mask the values within the key-value pairs of #1 with "[?]" to construct a schema (#2) that guides the LLMs in selectively extracting information from the document. Thereafter, we prompt the LLMs to compare the extracted structured information (#1 and #3), generate a concise analysis report, and utilize this report as a basis for further refining the summary content.

B Schema Design

```
{
    Subject Name: {
        {Key Name: Value Span},
        {Key Name: Value Span},
            ......
        {Key Name: Value Span}.
    }
}
```

Figure 6: We refer to the design of the Structured Extraction Language for Universal IE (Lu et al., 2022) and design our structured schema. Then, we provide an example from "Animal Farm" to demonstrate the extraction results.

Content

At first, Animal Farm prospers. Snowball works at teaching the animals to read, and Napoleon takes a group of young puppies to educate them in the principles of Animalism. When Mr. Jones reappears to take back his farm, the animals defeat him again, in what comes to be known as the Battle of the Cowshed, and take the farmer's abandoned gun as a token of their victory. As time passes, however, Napoleon and Snowball increasingly quibble over the future of the farm, and they begin to struggle with each other for power and influence among the other animals. Snowball concocts a scheme to build an electricity-generating windmill, but Napoleon solidly opposes the plan. At the meeting to vote on whether to take up the project, Snowball gives a passionate speech. Although Napoleon gives only a brief retort, he then makes a strange noise, and nine attack dogs—the puppies that Napoleon had confiscated in order to "educate"—burst into the barn and chase Snowball from the farm. Napoleon assumes leadership of Animal Farm and declares that there will be no more meetings. From that point on, he asserts, the pigs alone will make all of the decisions—for the good of every animal.

Extraction Result


```
{
  "Animal Farm": {
    "Initial State": "prospers",
    "Battle": "Battle of the Cowshed",
    "Victory Token": "farmer's abandoned gun"
  },
  "Snowball": {
    "Role": "teaches animals to read",
    "Plan": "build an electricity-generating windmill",
    "Speech": "gives a passionate speech",
    "Expulsion": "chased from the farm by attack dogs"
  },
  "Napoleon": {
    "Role": "educates young puppies in Animalism",
    "Opposition": "solidly opposes the windmill plan",
    "Action": "makes a strange noise to summon attack dogs",
    "Leadership": "assumes leadership of Animal Farm",
    "Meetings": "declares no more meetings",
    "Decision Making": "pigs will make all decisions"
  },
  "Mr. Jones": {
    "Attempt": "to take back his farm",
    "Outcome": "defeated in the Battle of the Cowshed"
  },
  "Attack Dogs": {
    "Origin": "puppies confiscated by Napoleon",
    "Action": "chase Snowball from the farm"
  }
}
```

C Prompt templates

C.1 Zero-shot

System prompt

Your objective is to Verify whether the [summary] is consistent with the factual content of the [document].
Key Points for Verification: (1) Core Content: Does the summary accurately reflect the key information and conclusions of the document? (2) Data and Facts: Are the data and facts in the summary consistent with those in the document?
Output:
Conclusion: Is it consistent?
Issues: If not consistent, identify the issues.
Suggestions: Provide recommendations for revision.

User prompt

Verify whether the [summary] accurately reflects the content of the [document].
<summary>. . . . . </summary>
<document>. . . . . </document>
Output [Conclusion, Issues and Suggestions]

C.2 Fine-tuning

System prompt

Your objective is to Verify whether the [summary] is consistent with the factual content of the [document]. Leverage your information extraction capabilities to identify the key structured information from both the summary and the document. The output format should be in the form of triplets (entity-relation-entity). Compare the differences in two key structured information.
Key Points for Verification: (1) Core Content: Does the summary accurately reflect the key information and conclusions of the document? (2) Data and Facts: Are the data and facts in the summary consistent with those in the document?
Output:
Conclusion: Is it consistent?
Issues: If not consistent, identify the issues.
Suggestions: Provide recommendations for revision.

User prompt

Verify whether the [summary] accurately reflects the content of the [document].
<summary>. . . . . </summary>
<document>. . . . . </document>
Output [Conclusion, Issues and Suggestions]

C.3 UIEFID

System prompt

Your objective is to Verify whether the [summary] is consistent with the factual content of the [document]. You should strictly follow the sequence below.
(1) Extract structured information from the summary and output it in JSON format. At the same time, replace all values in the JSON output with "[?]" to construct a schema for guided extraction.
(2) Strictly follow the structure of the schema to extract information from the document to fill in the "[?]" parts.
(3) Compare the structured information from the summary and the document, and answer the questions below.
Conclusion: Is it consistent?
Issues: If not consistent, identify the issues.
Suggestions: Provide recommendations for revision.

Note: The prompt here is significantly different from the one above and is related to the execution process of UIEFID. A three-turn dialogue prompting strategy should be adopted.

User prompt

User prompt (1st turn):
Extract structured information from the [summary] and construct the schema. The output result should be enclosed within the <Sum></Sum> and <Schema></Schema>tags.
<summary>. . . . . </summary>

User prompt (2nd turn):
Extract information from the document according to the structure of the [schema], and fill in the missing content marked by "[?]". The output result should be enclosed within the <Doc></Doc> tags.
<Schema>. . . . . </Schema>

User prompt (3rd turn):
Compare the structured information enclosed within the <Doc></Doc> and <Sum></Sum> tags, and output [Conclusion, Issues, and Suggestions].

D The Impact of Document Length

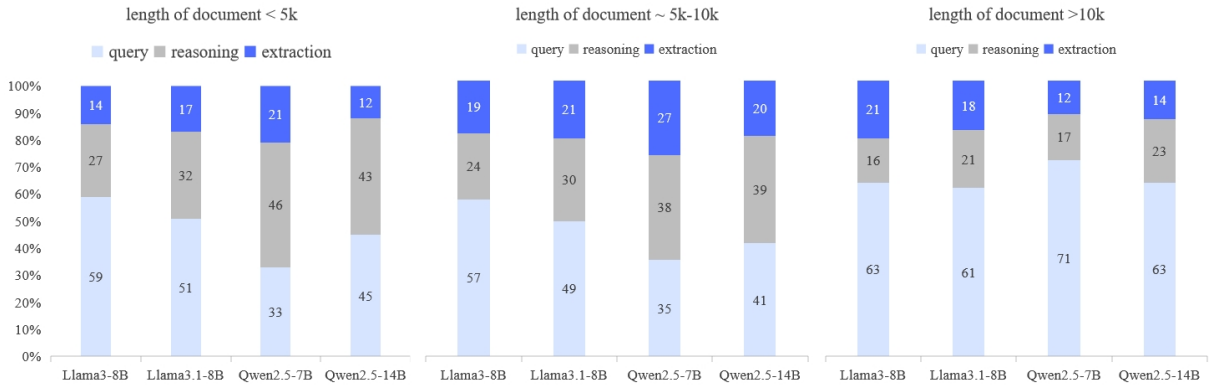

Figure 7: Statistical analysis of the tests reveals the average proportion of *query*, *reasoning*, and *extraction* segments during each detection process for open-source LLMs enhanced by UIEFID. Specifically, the ***query*** encompasses the prompt, summary, and document; ***reasoning*** refers to the segments that provide detection rationale and comparative analysis within the context of the response; and ***extraction*** corresponds to the parts related to the extraction of structured information. As document length increases, the relative proportion between reasoning and extraction gradually decreases, indicating that our approach effectively enhances the information extraction capabilities of LLMs while suppressing redundant reasoning.