Cognitive Mirroring for DocRE: A Self-Supervised Iterative Reflection Framework with Triplet-Centric Explicit and Implicit Feedback

Xu Han[1], Bo Wang[1]*, Yueheng Sun[1], Dongming Zhao[2], Zongfeng Qu[1,3], Ruifang He[1], Yuexian Hou[1]*, Qinghua Hu[1]

[1]College of Intelligence and Computing, Tianjin University, Tianjin, China
[2]AI Lab, China Mobile Communication Group Tianjin Co., Ltd.
[3]CHEARI (Beijing) Certification & Testing Co.,Ltd., Beijing, China

{hx_2001, bo_wang}@tju.edu.cn

Abstract

Large language models (LLMs) have advanced document-level relation extraction (DocRE), but DocRE is more complex than sentence-level relation extraction (SentRE), facing challenges like diverse relation types, coreference resolution and long-distance dependencies. Traditional pipeline methods, which detect relations before generating triplets, often propagate errors and harm performance. Meanwhile, fine-tuning methods require extensive human-annotated data, and in-context learning (ICL) underperforms compared to supervised approaches. We propose an iterative reflection framework for DocRE, inspired by human non-linear reading cognition. The framework leverages explicit and implicit relations between triplets to provide feedback for LLMs refinement. Explicit feedback uses logical rules-based reasoning, while implicit feedback reconstructs triplets into documents for comparison. This dual-process iteration mimics human semantic cognition, enabling dynamic optimization through self-generated supervision. For the first time, this achieves zero-shot performance comparable to fully supervised models. Experiments show our method surpasses existing LLM-based approaches and matches state-of-the-art BERT-based methods[1].

1 Introduction

DocRE aims to identify entity pairs and their semantic relations within long contexts, playing a vital role in various downstream NLP applications. LLMs have made significant progress in classical information extraction tasks (Xu et al., 2024). Recent studies leverage their strong instruction-following abilities and rich intrinsic knowledge to enhance DocRE performance (Ozyurt et al., 2024; Sun et al., 2024; Li et al., 2023).

However, DocRE is more challenging than SentRE due to the diversity of relation types, coreference resolution and long-distance dependencies within the document. Consequently, many SentRE methods (Wadhwa et al., 2023; Wan et al., 2023) cannot be directly applied to DocRE. To address these challenges, existing methods typically adopt a linear pipeline framework (Wei et al., 2024), which sequentially detects relations and generates triplets. However, this often leads to error propagation, degrading downstream performance. Based on the linear framework, these methods primarily rely on two paradigms: supervised methods requiring costly human annotations, and ICL approaches that underperform their supervised counterparts.

Existing linear frameworks do not align with the human cognitive process, which is iterative and conflict-driven rather than linear. The DocRE task inherently mirrors this process: when extracting relations from documents, humans naturally re-read ambiguous parts and resolve conflicts (aligned with the Construction-Integration Model (Kintsch, 1988) and Cognitive Dissonance Theory (Festinger, 1957)), and distinguish at both explicit and implicit levels (as per Dual Process Theory (Evans, 2003)). These theories suggest that text comprehension involves dynamic re-evaluation of earlier content when conflicts or missing information arise, not only emphasizing the iterative nature of human reading cognition, but also highlighting that the main driver of this iteration is the resolution of conflicts between knowledge. However, existing linear DocRE models violate these theories by processing documents unidirectionally without iterative verification, leading to suboptimal performance.

This disconnect becomes evident when considering the demonstrated success of the self-correction mechanism in other NLP domains (Pan et al., 2024), whose potential for DocRE remains strikingly underexplored. The absence of such reflective capabilities in existing approaches not only violates

*Corresponding authors.

[1]The authors are non-native English speakers, and AI assistants were used to polish certain sections of the paper, but were not used in research or coding.

Proceedings of the 1st Joint Workshop on Large Language Models and Structure Modeling (XLLM 2025), pages 197–217

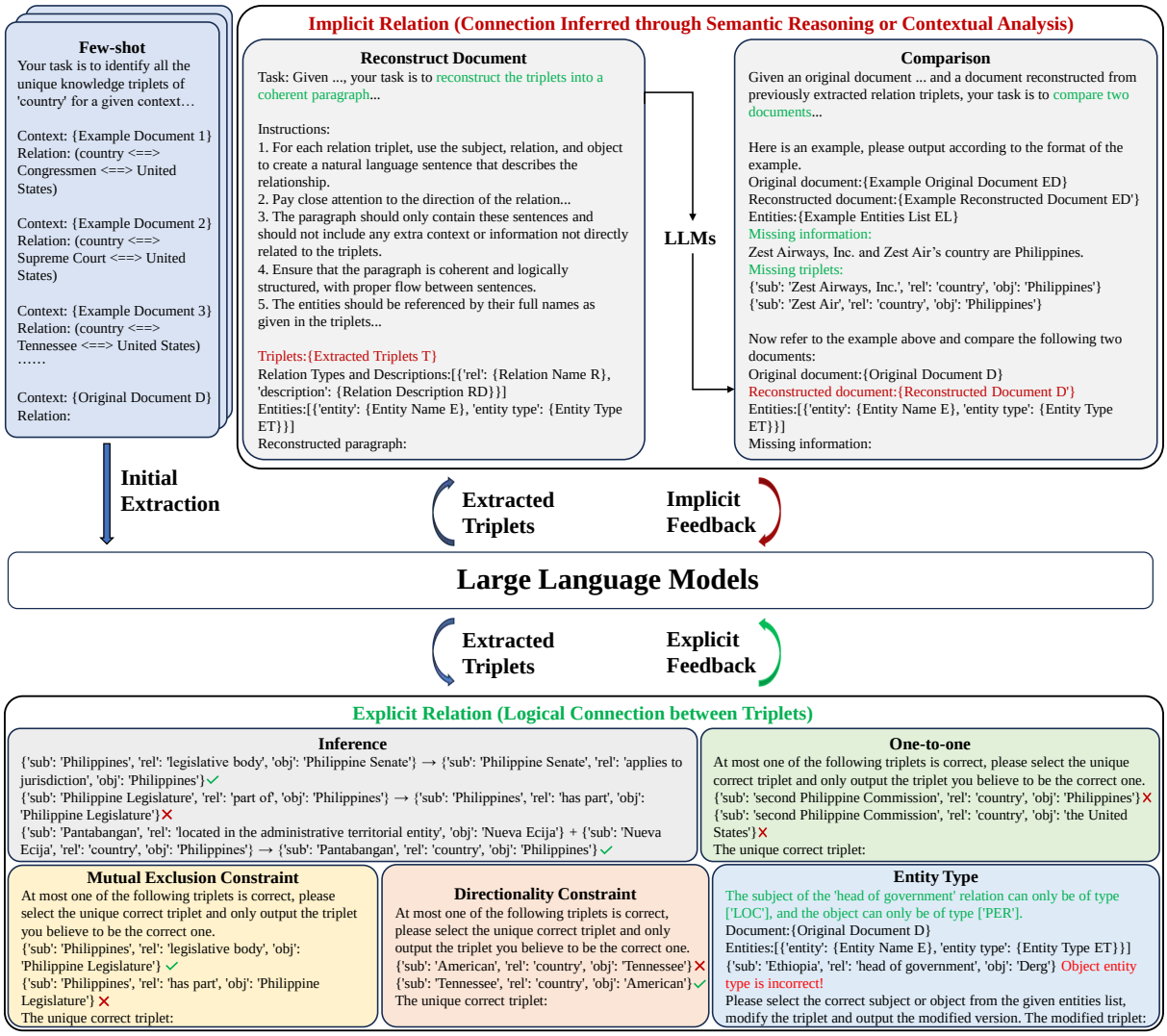

Figure 1: Overview of our framework, which consists of three modules: 1) Initial Extraction. 2) Obtain feedback on implicit relations between triplets through reconstruction (indicated by the red arrow). 3) Obtain feedback on explicit relations between triplets through logical rules-based inference and filtering (indicated by the green arrow).

cognitive theories but also limits their ability to handle the diverse relation types and coreference resolution of document-level understanding.

To bridge these gaps, we introduce an iterative reflection framework grounded in cognitive theories. By integrating self-supervised feedback signals, this framework circumvents the resource-intensive demands of supervised methods. To achieve comparable performance to supervised methods, we further draw on cognitive theories to introduce explicit and implicit relations between triplets, providing the corresponding self-supervised and low-cost rules-based feedback for LLMs refinement.

Specifically, our iterative reflection framework for ICL-based DocRE employs the self-correction mechanism, where LLMs follow an iterative extraction-verification-feedback-correction process to mitigate error propagation through constant error detection and correction. We define the implicit relation as a connection between triplets that is not explicitly stated but can be inferred through semantic reasoning or contextual analysis. Feedback on implicit relations is generated by reconstructing the extracted triplets into a document and comparing it with the original. In contrast, the explicit relation represents a direct logical connection between triplets. Feedback on explicit relations is provided through logical rules-based inference and filtering. By emphasizing and analyzing the relations between triplets, our method enhances contextual understanding and improves extraction accuracy, better handling the complex DocRE task. The overall framework is illustrated in Figure 1.

Our main work and contributions are as follows:

- Inspired by human non-linear reading cognition, we propose cognitive mirroring, a self-supervised iterative reflection framework for general-domain ICL-based DocRE that eliminates the need for supervised fine-tuning. To the best of our knowledge, this is the first exploration of self-correction in DocRE, enabling iterative extraction correction through feedback to mitigate error propagation.
- Drawing insights from the Dual Process Theory, we introduce the concepts of explicit and implicit relations between triplets. Triplet-centric explicit feedback is based on supplementary logical rules, while implicit feedback is generated through reconstruction. By jointly analyzing explicit logical relations and their interactions within the document and semantics, LLMs can better understand and reason through complex contexts, which is more suitable for the challenging DocRE task.
- Extensive experiments on DocRED and Re-DocRED datasets show our method surpasses existing LLM-based approaches and matches SOTA BERT-based methods.

2 Related Work

Human Cognitive Science Cognitive theories provide key insights into dual-process iteration. Construction-Integration Model (Kintsch, 1988) highlights reading as a dynamic process, where readers iteratively revise their understanding in response to conflicts or missing information. This aligns with the Cognitive Dissonance Theory (Festinger, 1957), which emphasizes the importance of resolving contradictions through reflective correction. Dual-Process Theory (Evans, 2003) suggests that human cognition involves both fast, intuitive processing (System 1) and slower, controlled reasoning (System 2). These theories inform our approach, integrating non-linear iteration and dual-relation analysis to enhance DocRE.

LLMs for DocRE Recent studies have explored approaches to harness the potential of LLMs for DocRE. For instance, (Sun et al., 2024) proposes a framework that employs zero-shot learning by generating synthetic data through a chain-of-retrieval prompt. Since DocRE involves many relation types and the output of LLMs is uncontrollable, (Li et al., 2023) combines LLMs with a natural language inference module to generate triplets, thus improving performance. REPLM (Ozyurt et al., 2024) introduces an in-context few-shot method leveraging pre-trained models, where triplets are generated based on relations and filtered according to the joint probabilities of entity pairs. Both DiVA-DocRE (Wu et al., 2024a) and AutoRE (Xue et al., 2024) first identify the relation types present in the document. AutoRE extracts the head entities for each relation before generating the complete triplets, while DiVA-DocRE incorporates active and passive voice information during extraction. However, these methods either require additional fine-tuning, or exhibit suboptimal performance when relying solely on ICL. Furthermore, decomposing DocRE into multiple steps may lead to error propagation, where errors in earlier stages negatively impact the performance of subsequent stages.

LLMs with Self-Correction Mechanism The self-correction mechanism for LLMs has demonstrated success across diverse tasks, including hallucination detection (Dhuliawala et al., 2024; Xue et al., 2023), mathematical reasoning (Zheng et al., 2024; Wu et al., 2024b; Jiang et al., 2024; Xue et al., 2023), question answering (Shinn et al., 2023; Zhao et al., 2023), dialogue generation (Madaan et al., 2023) and code optimization (Chern et al., 2023). However, applying this mechanism to DocRE remains underexplored. Metacognitive prompting (MP) (Wang and Zhao, 2024) enhances natural language understanding through structured self-aware evaluations by drawing on intrinsic knowledge and new insights. However, MP has only been preliminarily explored in the biomedical domain and does not involve iterative learning for LLMs. STAR (Ma et al., 2024) employs self-correction for data optimization and augmentation, while SRVF (Li et al., 2024) mitigates the bias of LLMs toward relation types through supervised feedback. Chem-FINESE (Wang et al., 2024) proposes a self-validation module for chemical entity extraction, employing contrastive loss to reduce excessive copying during extraction. However, these methods require additional fine-tuning and have not been extensively explored in DocRE. In this study, we propose a self-correction framework for general-domain DocRE that requires no supervised fine-tuning and enables iterative error correction, bridging the gap between the self-correction mechanism and DocRE.

3 Methodology

As illustrated in Figure 1, the proposed framework generates feedback for iterative modification of

LLMs by analyzing explicit and implicit relations between triplets. Specifically, the framework consists of three phases: 1) initial extraction, 2) triplet-centric implicit feedback through reconstruction, and 3) explicit feedback based on logical rules.

3.1 Task Formulation

DocRE aims to predict relations between entity pairs that may appear across multiple sentences in a document. Given a document D with entities $E = \{e_i\}$, the goal is to identify relation triplets (r, e_s, e_o), where $r \in R$, $e_s, e_o \in E$, and R is the set of predefined relation types, e_s and e_o respectively denote the subject entity and the object entity. Each entity may be mentioned multiple times.

3.2 Initial Extraction

We build upon the REPLM framework (Ozyurt et al., 2024), which generates triplets for each relation in the relation list and filters these triplets by calculating the joint probabilities of subject-object pairs. Finally, the framework aggregates predictions from multiple retrieved demonstrations to obtain the final predictions.

3.3 Triplet-Centric Implicit Feedback

Implicit Relation We define implicit relations between triplets as the connections that are not directly reflected in explicit expression but can be inferred through semantic reasoning or contextual analysis. These relations encompass the mutual influence of triplets on each other when they serve as context, as well as semantic connections between triplets within a document that cannot be clearly represented by formal definitions. For instance, in the text *It meets at Legislative Hall in Dover, Delaware, convening on the second Tuesday of January of odd years*, the triplets *(Legislative Hall, located in the administrative territorial entity, Dover)* and *(Legislative Hall, located in the administrative territorial entity, Delaware)* do not explicitly state the geographical relation between *Dover* and *Delaware*. However, contextual inference reveals that *Dover* is a location within *Delaware*, establishing an implicit relation between these two triplets. Although the relation is not explicitly stated, it is inferred by semantic reasoning.

Drawing on cognitive science principles of human reverse verification, our framework generates feedback on implicit relations between triplets by leveraging extracted triplets to reconstruct a document and comparing the reconstructed document with the original document at a fine-grained level.

For DocRE, the reconstruction process serves as a representation of extraction quality, where better extraction results yield smaller discrepancies between reconstructed and original documents. This comparative analysis enables error identification through two primary mechanisms: 1) detection of missing content when original document elements are absent in the reconstructed version, indicating potential extraction failures, and 2) identification of semantic contradictions in the reconstructed document, revealing extraction errors.

The rationale for employing document-level comparison rather than direct triplet-document comparison is twofold. First, LLMs excel at understanding and processing unstructured data, making document-level comparison more effective for identifying discrepancies. Second, reconstructing all extracted triplets into a single document facilitates the integration and analysis of implicit relations between triplets, whereas direct comparison typically examines triplets in isolation, potentially overlooking contextual information.

3.4 Triplet-Centric Explicit Feedback

Explicit Relation Our method defines explicit relations as logical connections between triplets, which can be inferred and filtered through rules to identify which triplets have not been extracted, which extracted triplets are incorrect, or which are contradictory to each other. These analyses provide feedback for LLMs to correct their predictions. The rules-based module we implement comprises five principal categories:

Inference We follow MILR (Fan et al., 2022), a logic-enhanced framework designed to enhance DocRE by mining and injecting logical rules. It mines logical rules from annotations based on frequencies. For example:

$$\textit{father (x, y)} \wedge \textit{spouse (y, z)} \rightarrow \textit{mother (x, z)}$$
$$\textit{has part (x, y)} \rightarrow \textit{part of (y, x)}$$

We utilize inference rules to identify missing or incorrectly extracted triplets.

One-to-one For certain relation types, a subject can maintain the relation with only one object. For instance, *a city can only belong to one country, and a person can only have one date of birth*. If the extracted triplets contain the following two triplets, it can be determined that at most one of these triplets is correct, since *Beibu Gulf Economic Rim can only belong to one country*. Then, LLMs are guided

to select the most plausible triplet based on the document and relation description.

(Beibu Gulf Economic Rim, country, China)
(Beibu Gulf Economic Rim, country, Vietnam)

Mutual Exclusion Constraint This constraint prohibits entities from simultaneously maintaining mutually exclusive relations. For instance, the following two triplets, *Lansing is the capital of Michigan* and *Michigan is a direct subdivision of Lansing* are logically contradictory, as they attempt to establish both *contains administrative territorial entity* and *capital of* relations between the same entity pair. Such mutually exclusive relations cannot coexist, ensuring that at most one of the conflicting triplets can be correct. The framework identifies these contradictions and guides LLMs to select.

(Lansing, contains administrative territorial entity, Michigan)
(Lansing, capital of, Michigan)

Directionality Constraint This constraint emphasizes the significance of relation directionality between entities, prohibiting reversals that result in logical inconsistencies. For example, consider the triplet *(ITS, developer, SpaceX)*, which indicates that *SpaceX is the developer of ITS*. This relation is inherently unidirectional, reversing the triplet to *(SpaceX, developer, ITS)* creates a logical contradiction, as *ITS cannot be the developer of SpaceX* if *SpaceX is already the developer of ITS*. Consequently, at most one of the following two triplets can be correct, and the framework identifies such directional violations and guides LLMs to select.

(ITS, developer, SpaceX)
(SpaceX, developer, ITS)

Entity Type To mitigate the directional errors exhibited by LLMs, we comprehensively redefined the relation descriptions of datasets to emphasize not only the directionality of relations but also the potential content of the subject and object entities. Meanwhile, we also consider type constraints for both the subject and object of each relation. For example, the subject of the *head of government* relation must be of type *location*, and the object must be of type *person*, indicating that *object is the head of government of subject*. Triplets violating these type constraints are identified as erroneous and used as feedback for LLMs refinement. Examples of relation descriptions and entity type constraints are presented in Appendix A.

By iteratively generating feedback based on both explicit and implicit relations between triplets, LLMs can continuously optimize the extraction results and improve performance. Our method is not only suitable for DocRE that requires complex text processing and judgment but also helps mitigate error propagation by constantly identifying and correcting errors.

The method can be implemented with only the definition of relations and logical rules, without the need for any oracle labels. For initial extraction, we adopt the five-shot prompts from REPLM (Ozyurt et al., 2024). During iterations, we employ one-shot prompts for reconstructing and identifying unextracted triplets, while maintaining zero-shot prompts for other steps. This implementation avoids the issue raised by (Huang et al., 2024), regarding sub-optimal prompts for generating initial responses, while providing more informative instructions about the task in the feedback prompts.

The full prompts of entire framework are presented in Appendix B.

4 Experiments

4.1 Datasets and Evaluation Metric

We conduct experiments on DocRED (Yao et al., 2019) and Re-DocRED (Tan et al., 2022), two large-scale crowd-sourced benchmark datasets tailored for DocRE. Due to the absence of ground truth labels in the test set of DocRED, we perform evaluations only on the dev set. Further details are provided in Appendix C.

Our evaluation employs the strict Micro F1 metric, where a prediction is considered correct only when it precisely identifies both the subject and object entities along with their relation. It's important to highlight that within the datasets, multiple entity mentions may refer to the same underlying entity. Therefore, predictions matching any alias of the annotated entity are accepted as correct. To ensure a rigorous and valid evaluation, regardless of the number of aliases an entity possesses, it will only be counted once in the triplet alignment evaluation. All incorrect predictions are flagged as false positives. This approach ensures a precise and statistically valid evaluation, lending robust credibility to our results.

4.2 Implementation

We employ GPT-3.5-Turbo, GPT-4o, ChatGLM3-6B, and LLaMA3-8B as our backbone LLMs. To obtain deterministic outputs, we set a low temper-

ature, such as 0.001, while keeping all other parameters at their default values. The total API cost for GPT-3.5-Turbo and GPT-4o used in our exploration and experiments is approximately $1,000. ChatGLM3-6B is deployed on an NVIDIA Tesla P100 PCI-E 16GB GPU, and LLaMA3-8B is deployed via Ollama.

In the REPLM framework (Ozyurt et al., 2024), $L = 5$ is set to obtain five sets of in-context demonstrations, each of which is used to extract triplets individually before aggregation. In our experiments, we set $L = 1$ to initially extract triplets only once, while keeping all other parameter configurations identical to those in the REPLM framework.

4.3 Baselines

We compare our framework with both BERT-based and LLM-based methods on the datasets. The BERT-based baselines, recognized for achieving SOTA performance, leverage BERT family pretrained models as encoders. The LLM-based baselines incorporate techniques such as supervised fine-tuning and chain-of-thought (CoT) to enhance relation extraction performance. All the baselines we selected for comparison are shown in Table 1 with detailed descriptions provided in Appendix D.

5 Results and Discussion

5.1 Main Results

Due to the iterative framework involving repeated extraction and cost constraints, the main results of our method are based on a single run, as presented in Table 1. From these results, we can draw the following conclusions.

Our framework demonstrates significant performance improvements across two datasets. Specifically, our method achieves SOTA performance with a micro F1 score of 69.58 on the DocRED dev set using GPT-4o, surpassing previous BERT-based methods by 1.45 and prior LLM-based methods by 2.11. On the Re-DocRED dev set, where SOTA methods were not evaluated, our method outperforms all LLM-based methods, including those with fine-tuning. On the Re-DocRED test set, our method significantly outperforms recent LLM-based methods and narrows the performance gap with SOTA methods. Although it does not achieve SOTA performance on the Re-DocRED test set, the results after two iterations show significant improvements over the initial extraction across both datasets. For example, on the Re-DocRED dataset, the performance improvement is 24.77 on the dev set and approximately 22.72 on the test set, both based on GPT-4o, demonstrating the effectiveness of our method.

Our framework outperforms the self-consistency method. The results in (Huang et al., 2024) reveal that some self-correction methods do not outperform self-consistency. As shown in Table 1, REPLM (Ozyurt et al., 2024) uses five sets of in-context demonstrations to extract five times and filters triplets based on the probabilities of entity pairs. This can be considered as a self-consistency method that establishes five reasoning paths. In contrast, our method achieves better performance with only two iterations. Specifically, based on GPT-4o, our method reaches an F1 score of 69.58 after two iterations, compared to REPLM's 67.47. This demonstrates that our method not only surpasses self-consistency in performance but also achieves higher efficiency with lower resource demands. We provide detailed token usage statistics for DocRED's 998 documents (calculated using OpenAI's tiktoken) in Appendix E, demonstrating the practical advantages of our framework.

Our framework is applicable to a wide range of LLMs and demonstrates enhanced performance with more powerful models. To validate the adaptability and robustness of our method for DocRE across different LLMs, we conduct experiments with four representative LLMs: ChatGLM3-6B, LLaMA3-8B, GPT-3.5, and GPT-4o, covering both open-source and closed-source models, as well as varying model sizes. The results confirm the applicability of our method across different LLMs and its effectiveness in handling the DocRE task. Whether for the initial extraction or after two iterations, GPT-4o consistently outperforms GPT-3.5, followed by LLaMA3-8B and ChatGLM3-6B. Although LLaMA3-8B starts with relatively low F1 scores, its performance improves significantly after two iterations, surpassing that of ChatGLM3-6B. While ChatGLM3-6B and LLaMA3-8B do not outperform fine-tuned LLM-based baselines, our zero-shot self-supervised method substantially narrows the performance gap. Iterative improvements further validate the effectiveness of our approach. In addition, more powerful LLMs exhibit greater performance gains through iteration. For example, on the DocRED dev set, the F1 score of GPT-3.5 increased by 19.45, while GPT-4o improved by 23.17. This shows that our framework not only adapts ef-

Method	PLM	DocRED	Re-DocRED	
			dev	test
BERT-based				
JMRL-DREEAM (Qi et al., 2024)	$\text{RoBERTa}_{\text{large}}$	67.61	-	78.61
DREEAM (Ma et al., 2023)	$\text{RoBERTa}_{\text{large}}$	67.41	-	<u>81.44</u>
DocRE-CLiP (Jain et al., 2024)	$\text{BERT}_{\text{base}}$	<u>68.13</u>	-	**81.55**
LLM-based				
GenRDK (Sun et al., 2024)	LLaMA2-13B-CHAT	42.50	39.90	41.30
$\text{DiVA-DocRE}^{\text{GT}}$ (Wu et al., 2024a)	LLama3-7B	55.48	<u>61.99</u>	61.40
REPLM (Ozyurt et al., 2024)	GPT-3.5	59.66	$41.07^{\dagger}$	$40.30^{\dagger}$
	GPT-4o	67.47	$41.67^{\dagger}$	$41.48^{\dagger}$
AutoRE (Xue et al., 2024)	Vicuna-7B	-	54.29	53.84
Our Framework				
Initial Extraction	ChatGLM3-6B	21.30	18.18	18.29
	LLama3-8B	9.40	6.97	6.71
	GPT-3.5	43.13	39.32	37.71
	GPT-4o	46.41	40.02	40.55
Iteration 2	ChatGLM3-6B	43.71	34.99	35.38
	LLama3-8B	53.86	43.83	43.01
	GPT-3.5	62.58	58.48	56.95
	GPT-4o	**69.58**	**64.79**	63.27

Table 1: Results on the DocRED and Re-DocRED datasets. Shown: Micro F1 scores. The results of all baseline methods are taken from their papers, while † denotes results reproduced using the official code provided by the methods. For each dataset, the best result is in **bold**, while the second-best result is <u>underlined</u>.

fectively to various LLMs but also benefits more significantly from stronger model capabilities.

5.2 Analysis on Relation Density

Since our method considers explicit and implicit relations between triplets, and the relations between triplets are intuitively closely related to the relation density, we conduct experiments to evaluate its effectiveness on datasets with varying relation densities. Documents in each dataset are arranged in descending order of relation density and divided into two equally sized subsets: one with high relation density and the other with low relation density. We define relation density as the ratio of the number of triplet facts to the number of entity pairs. Formally, relation density is computed as follows:

$$relation\_density = \frac{tn}{epn} \quad (1)$$

$$epn = \frac{en * (en - 1)}{2} \quad (2)$$

where tn and en represent the number of triplets and entities annotated in the document, respectively, and epn is the number of entity pairs.

Our method demonstrates superior performance improvements on datasets with high relation density. As shown in Table 2, across both datasets, the performance improvements after two iterations are greater for high relation density datasets compared to low relation density datasets. Although the initial F1 scores on high relation density datasets are lower due to the increased complexity of extraction in such datasets, they surpass those of low relation density datasets after two iterations on the dev sets of DocRED and Re-DocRED. Notably, on these dev sets, the performance on low relation density datasets even surpasses that of baseline methods, though it remains slightly lower on Re-DocRED test set. This indicates that our method effectively captures relational information and performs better when the relations between triplets are more tightly connected, particularly in scenarios requiring complex relational parsing within document-level context. For datasets with lower relation density, methods such as information summarization could be employed to increase

Dataset	DocRED		Re-DocRED			
	dev		dev		test	
Overall F1	46.41	69.58 (+23.17)	40.02	64.79 (+23.12)	40.55	63.27 (+21.79)
High Relation Density F1	45.98	70.63 (+24.65)	39.41	65.09 (+25.68)	39.29	63.04 (+23.75)
Low Relation Density F1	46.85	68.53 (+21.68)	40.63	64.50 (+23.87)	41.81	63.50 (+21.69)

Table 2: Analysis on relation density. For each set, the left column displays the micro F1 scores of the initial extraction, while the right column shows the F1 scores after two iterations. The values in parentheses represent the increase in the F1 scores after two iterations. All these results are based on GPT-4o.

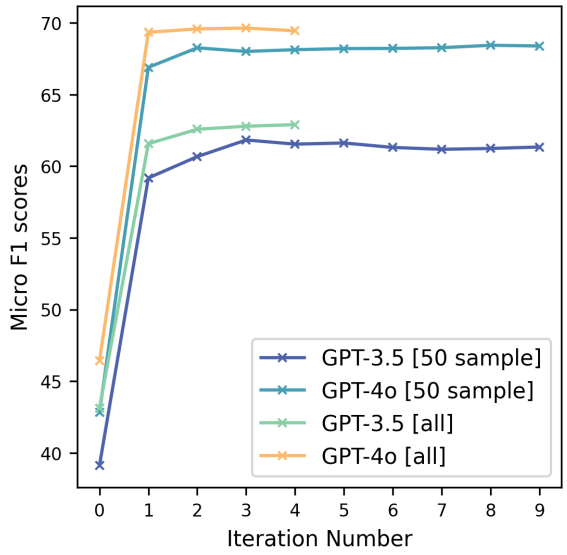

Figure 2: Impact of iteration number on DocRED dev set.

density, which can be explored in future work.

5.3 Impact of Iteration Number

The number of interactions refers to the frequency with which the LLMs receive feedback and improve the extraction results. The zeroth iteration represents the initial extraction. Given the high cost of API calls, we randomly sample 50 documents from the DocRED dev set and conduct 10 iterations. For the entire DocRED dev set, we perform 5 iterations using both GPT-3.5 and GPT-4o. As illustrated in Figure 2, our results indicate that the model reaches its performance peak during the second or third iteration, after which the performance fluctuates around the peak.

The diminishing returns in performance improvements in subsequent iterations may be attributed to several factors. First, the reconstruction of implicit relations faces the following challenges: 1) Certain triplets involve deep semantic understanding and multi-step reasoning, such as identifying hierarchical relations among entities, are inherently difficult for the model to detect. 2) Entities with low frequencies are less likely to be identified by LLMs. 3) Previous research (Wadhwa et al., 2023) has indicated that evaluating LLM-based methods should not rely exclusively on exact matches to target triplets. LLMs can generate outputs that are semantically proximate but fail to meet exact-matching criteria. Specially, t is a triplet in the annotations and t' is a semantically similar triplet generated by the LLMs that does not satisfy the exact-matching requirement, the reconstructed document will contain the corresponding semantic content. Consequently, t will not be considered unextracted, and t' will not be identified as an incorrect triplet to remove. 4) Issues with the directionality of relations may also confuse the semantics of reconstructed documents. Additionally, performance bottlenecks may arise due to logical reasoning based on rules, which can lead to logical loops.

The observed performance degradation during iterations can be attributed to two main factors: 1) Reconstruction and logical rules-based inference may identify unextracted triplets that are not included in annotations, resulting in a decrease in precision. 2) Incorrect triplets introduced by reconstruction and logical inference can affect the selection of LLMs during subsequent filtering with one-to-one, mutual exclusion, and directionality constraints, resulting in a decrease in recall.

5.4 Ablation study

Figure 3 illustrates the performance improvement process of two datasets during two iterations, starting from initial extraction. As shown, both the reconstruction module for implicit relations and the module for explicit relations contribute to performance enhancements to varying degrees. As mentioned above, although rules-based inference and filtering may form a closed loop, and reconstruction may introduce incorrect triplets, the synergistic combination of both modules yields superior performance in the second iteration compared to

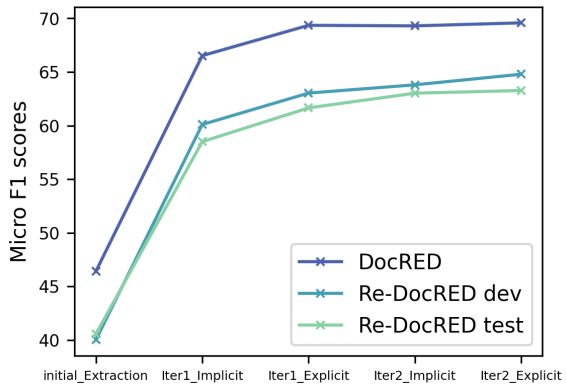

Figure 3: Ablation study. All the results are based on GPT-4o.

the first. This indicates the effectiveness of both modules within the overall framework.

A detailed case study of a complete iteration process is presented in Appendix F.

6 Conclusion

Inspired by human non-linear reading cognition, we propose an iterative reflection framework for ICL-based DocRE that validates both explicit and implicit relations between triplets. The framework initiates with preliminary extraction and then establishes an iterative extraction-verification-feedback-correction procedure. Explicit relational feedback is based on logical rules-based reasoning, while feedback on implicit relations is generated by reconstructing the extracted triplets into a document and comparing it with the original. This dual-process mechanism mimics human cognition, enabling self-supervised optimization without reliance on annotated data. Extensive experiments demonstrate that our framework not only narrows the performance gap with SOTA BERT-based methods, but also performs better in complex DocRE task. Additionally, it helps mitigate error propagation by continuously identifying and correcting errors. Furthermore, our method explores the application of the self-correction mechanism in DocRE, offering a more efficient and convenient solution. It provides new insights and directions for the application of LLMs in DocRE and offers a valuable reference for future exploration.

Limitations

Due to cost constraints and limitations in biomedicine domain knowledge, we have not yet conducted experiments on CDR, GDA and DocGNRE datasets, nor have we explored the performance on multilingual datasets. Additionally, we have not compared multiple prompt variants. We have not explored the impact of using few-shot prompts in all modules or considered the influence of the order of in-context learning demonstrations, instructions, and relation descriptions in the prompts on experimental results, nor have we accounted for the impact of factors like demonstrations quality. These limitations will be explored in our future work.

Moreover, the generation of Triplet-Centric Explicit Feedback in our framework currently relies on manually defined logical rules. Although our framework only requires low-cost domain-specific rule definitions, either for general or specialized domains, this dependence still introduces some level of manual effort. We will explore the automatic induction or learning of more detailed, accurate, and dataset-specific logical rules to further enhance the performance and generalizability of our method.

Ethics Statement

All documents and models used in this study were obtained from open-source sources, ensuring transparency and accessibility. Our framework provides an effective solution for DocRE using LLMs, without requiring additional training or fine-tuning. This makes it easier to deploy and use in practice. However, we acknowledge the potential risk of misuse, particularly regarding the extraction of personal or sensitive information. To mitigate this concern, we only use public benchmark datasets DocRED and Re-DocRED for evaluation. These datasets do not involve personal privacy. We also advocate not applying our framework to extract or analyze any private data without user authorization.

Moreover, we recognize that LLMs may inherit implicit biases from their training data. Although our framework can identify triplets, biased or unintended outputs may still occur, especially when analyzing sensitive relation types or entities. We advocate for the careful and responsible use of such models and encourage further research to identify, evaluate and mitigate these potential biases.

Acknowledgments

This work was supported by the National Natural Science Foundation of China (62376188, 62272340, 62276187, 62376192, 62166022) and the Key Technology Research and Industrial Ap-

plication Demonstration of General Large Model with Autonomous Intelligent Computing Power, No.24ZGZNGX00020).

References

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

Tim Dettmers, Artidoro Pagnoni, Ari Holtzman, and Luke Zettlemoyer. 2023. Qlora: efficient finetuning of quantized llms. In *Proceedings of the 37th International Conference on Neural Information Processing Systems*, NIPS '23, Red Hook, NY, USA. Curran Associates Inc.

Shehzaad Dhuliawala, Mojtaba Komeili, Jing Xu, Roberta Raileanu, Xian Li, Asli Celikyilmaz, and Jason Weston. 2024. Chain-of-verification reduces hallucination in large language models. In *Findings of the Association for Computational Linguistics: ACL 2024*, pages 3563–3578, Bangkok, Thailand. Association for Computational Linguistics.

Jonathan St.B.T. Evans. 2003. In two minds: dual-process accounts of reasoning. *Trends in Cognitive Sciences*, 7(10):454–459.

Shengda Fan, Shasha Mo, and Jianwei Niu. 2022. Boosting document-level relation extraction by mining and injecting logical rules. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 10311–10323, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Leon Festinger. 1957. *A Theory of Cognitive Dissonance*. Stanford University Press, Redwood City.

Jie Huang, Xinyun Chen, Swaroop Mishra, Huaixiu Steven Zheng, Adams Wei Yu, Xinying Song, and Denny Zhou. 2024. Large language models cannot self-correct reasoning yet. In *The Twelfth International Conference on Learning Representations*.

Monika Jain, Raghava Mutharaju, Ramakanth Kavuluru, and Kuldeep Singh. 2024. Revisiting document-level relation extraction with context-guided link prediction. In *Proceedings of the Thirty-Eighth AAAI Conference on Artificial Intelligence and Thirty-Sixth Conference on Innovative Applications of Artificial Intelligence and Fourteenth Symposium on Educational Advances in Artificial Intelligence*, AAAI'24/IAAI'24/EAAI'24. AAAI Press.

Weisen Jiang, Han Shi, Longhui Yu, Zhengying Liu, Yu Zhang, Zhenguo Li, and James Kwok. 2024. Forward-backward reasoning in large language models for mathematical verification. In *Findings of the Association for Computational Linguistics: ACL 2024*, pages 6647–6661, Bangkok, Thailand. Association for Computational Linguistics.

Walter Kintsch. 1988. The role of knowledge in discourse comprehension: a construction-integration model. *Psychological review*, 95 2:163–82.

Junpeng Li, Zixia Jia, and Zilong Zheng. 2023. Semi-automatic data enhancement for document-level relation extraction with distant supervision from large language models. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 5495–5505, Singapore. Association for Computational Linguistics.

Yongqi Li, Xin Miao, Shen Zhou, Mayi Xu, Yuyang Ren, and Tieyun Qian. 2024. Enhancing relation extraction via supervised rationale verification and feedback. *Preprint*, arXiv:2412.07289.

Mingyu Derek Ma, Xiaoxuan Wang, Po-Nien Kung, P. Jeffrey Brantingham, Nanyun Peng, and Wei Wang. 2024. Star: boosting low-resource information extraction by structure-to-text data generation with large language models. In *Proceedings of the Thirty-Eighth AAAI Conference on Artificial Intelligence and Thirty-Sixth Conference on Innovative Applications of Artificial Intelligence and Fourteenth Symposium on Educational Advances in Artificial Intelligence*, AAAI'24/IAAI'24/EAAI'24. AAAI Press.

Youmi Ma, An Wang, and Naoaki Okazaki. 2023. DREEAM: Guiding attention with evidence for improving document-level relation extraction. In *Proceedings of the 17th Conference of the European Chapter of the Association for Computational Linguistics*, pages 1971–1983, Dubrovnik, Croatia. Association for Computational Linguistics.

Aman Madaan, Niket Tandon, Prakhar Gupta, Skyler Hallinan, Luyu Gao, Sarah Wiegreffe, Uri Alon, Nouha Dziri, Shrimai Prabhumoye, Yiming Yang, Shashank Gupta, Bodhisattwa Prasad Majumder, Katherine Hermann, Sean Welleck, Amir Yazdanbakhsh, and Peter Clark. 2023. Self-refine: Iterative refinement with self-feedback. In *Thirty-seventh Conference on Neural Information Processing Systems*.

Yilmazcan Ozyurt, Stefan Feuerriegel, and Ce Zhang. 2024. Document-level in-context few-shot relation extraction via pre-trained language models. *Preprint*, arXiv:2310.11085.

Liangming Pan, Michael Saxon, Wenda Xu, Deepak Nathani, Xinyi Wang, and William Yang Wang. 2024. Automatically correcting large language models: Surveying the landscape of diverse automated correction strategies. *Transactions of the Association for Computational Linguistics*, 12:484–506.

Kunxun Qi, Jianfeng Du, and Hai Wan. 2024. End-to-end learning of logical rules for enhancing document-level relation extraction. In *Proceedings of the 62nd Annual Meeting of the Association for Computational*

Linguistics (Volume 1: Long Papers), pages 7247–7263, Bangkok, Thailand. Association for Computational Linguistics.

Noah Shinn, Federico Cassano, Ashwin Gopinath, Karthik Narasimhan, and Shunyu Yao. 2023. Reflexion: language agents with verbal reinforcement learning. In *Advances in Neural Information Processing Systems*, volume 36, pages 8634–8652. Curran Associates, Inc.

Qi Sun, Kun Huang, Xiaocui Yang, Rong Tong, Kun Zhang, and Soujanya Poria. 2024. Consistency guided knowledge retrieval and denoising in llms for zero-shot document-level relation triplet extraction. In *Proceedings of the ACM Web Conference 2024*, WWW '24, page 4407–4416, New York, NY, USA. Association for Computing Machinery.

Qingyu Tan, Lu Xu, Lidong Bing, Hwee Tou Ng, and Sharifah Mahani Aljunied. 2022. Revisiting DocRED - addressing the false negative problem in relation extraction. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 8472–8487, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Somin Wadhwa, Silvio Amir, and Byron Wallace. 2023. Revisiting relation extraction in the era of large language models. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 15566–15589, Toronto, Canada. Association for Computational Linguistics.

Zhen Wan, Fei Cheng, Zhuoyuan Mao, Qianying Liu, Haiyue Song, Jiwei Li, and Sadao Kurohashi. 2023. GPT-RE: In-context learning for relation extraction using large language models. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 3534–3547, Singapore. Association for Computational Linguistics.

Qingyun Wang, Zixuan Zhang, Hongxiang Li, Xuan Liu, Jiawei Han, Huimin Zhao, and Heng Ji. 2024. Chem-FINESE: Validating fine-grained few-shot entity extraction through text reconstruction. In *Findings of the Association for Computational Linguistics: EACL 2024*, pages 1–16, St. Julian's, Malta. Association for Computational Linguistics.

Yuqing Wang and Yun Zhao. 2024. Metacognitive prompting improves understanding in large language models. In *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pages 1914–1926, Mexico City, Mexico. Association for Computational Linguistics.

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. 2024. Chatie: Zero-shot information extraction via chatting with chatgpt. *Preprint*, arXiv:2302.10205.

Yiheng Wu, Roman Yangarber, and Xian Mao. 2024a. Diva-docre: A discriminative and voice-aware paradigm for document-level relation extraction. *Preprint*, arXiv:2409.13717.

Zhenyu Wu, Meng Jiang, and Chao Shen. 2024b. Get an a in math: progressive rectification prompting. In *Proceedings of the Thirty-Eighth AAAI Conference on Artificial Intelligence and Thirty-Sixth Conference on Innovative Applications of Artificial Intelligence and Fourteenth Symposium on Educational Advances in Artificial Intelligence*, AAAI'24/IAAI'24/EAAI'24. AAAI Press.

Derong Xu, Wei Chen, Wenjun Peng, Chao Zhang, Tong Xu, Xiangyu Zhao, Xian Wu, Yefeng Zheng, Yang Wang, and Enhong Chen. 2024. Large language models for generative information extraction: A survey. *Preprint*, arXiv:2312.17617.

Lilong Xue, Dan Zhang, Yuxiao Dong, and Jie Tang. 2024. Autore: Document-level relation extraction with large language models. *Preprint*, arXiv:2403.14888.

Tianci Xue, Ziqi Wang, Zhenhailong Wang, Chi Han, Pengfei Yu, and Heng Ji. 2023. Rcot: Detecting and rectifying factual inconsistency in reasoning by reversing chain-of-thought. *Preprint*, arXiv:2305.11499.

Yuan Yao, Deming Ye, Peng Li, Xu Han, Yankai Lin, Zhenghao Liu, Zhiyuan Liu, Lixin Huang, Jie Zhou, and Maosong Sun. 2019. DocRED: A large-scale document-level relation extraction dataset. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 764–777, Florence, Italy. Association for Computational Linguistics.

Ruochen Zhao, Xingxuan Li, Shafiq Joty, Chengwei Qin, and Lidong Bing. 2023. Verify-and-edit: A knowledge-enhanced chain-of-thought framework. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 5823–5840, Toronto, Canada. Association for Computational Linguistics.

Chuanyang Zheng, Zhengying Liu, Enze Xie, Zhenguo Li, and Yu Li. 2024. Progressive-hint prompting improves reasoning in large language models. In *AI for Math Workshop @ ICML 2024*.

A Relation Descriptions and Entity Type Constraints

Table 5 presents examples of relation descriptions and entity type constraints for DocRED and Re-DocRED datasets.

B Prompts

The initial extraction prompts follow REPLM (Ozyurt et al., 2024). Other prompts are presented from Table 6 to Table 12.

C Datasets

Our framework is evaluated on two English DocRE benchmark datasets, DocRED and Re-DocRED. More statistical information is listed in Table 3.

DocRED A large-scale human-annotated dataset derived from Wikipedia and Wikidata, plays a key role in DocRE. It features a comprehensive annotation schema that includes entity mentions, types, relational facts, and supporting evidence. With 96 predefined relation types, DocRED presents a rich and challenging environment, requiring multi-sentence reasoning for its relational facts. The dataset consists of three sets: a train set with 3053 documents, a dev set with 998 documents and a test set with 1000 documents. Since the test set lacks a ground truth file, evaluations are typically conducted on the dev set.

Re-DocRED While DocRED is a widely recognized benchmark, its annotations are incomplete, leading to false negatives. To address this, (Tan et al., 2022) introduces Re-DocRED, a revised version that supplements positive instances missing in DocRED. Re-DocRED includes a test set with 500 documents and a dev set with 500 documents, ensuring a more comprehensive and accurate assessment for DocRE task.

Dataset	Split	#Doc.	#Rel.	#Ent.	#Facts.
DocRED	train	3053	96	59493	38180
	dev	998		19578	12323
	test	1000		19539	-
Re-DocRED	train	3053	96	59359	85932
	dev	500		9684	17284
	test	500		9779	17448

Table 3: Statistics on datasets, where Doc. (resp. Rel. or Ent.) abbreviates documents (resp. relations or entities).

D Baselines

D.1 BERT-based Baselines

JMRL-DREEAM (Qi et al., 2024) A rule-enhanced DocRE framework that jointly models document-level relation extraction and logical rules in an end-to-end fashion. JMRL integrates the neural DocRE backbone (DREEAM) with a parameterized rule reasoning module that simulates logical inference. To effectively align the predictions from both components, the framework incorporates a residual connection mechanism and an auxiliary loss, enabling a better reconciliation between neural predictions and symbolic reasoning.

DREEAM (Ma et al., 2023) A memory-efficient model that incorporates evidence-guided supervision into attention mechanisms. DREEAM leverages evidence data to supervise the computation of entity-pair-specific context embeddings, encouraging higher attention weights on informative tokens without introducing additional trainable parameters. To address the lack of annotated evidence, they further propose a self-training strategy that learns entity resolution (ER) from automatically generated evidence on large-scale unlabeled data.

DocRE-CLiP (Jain et al., 2024) A knowledge-enhanced framework that reformulates DocRE as link prediction over a knowledge graph enriched with document-derived reasoning and external knowledge from Wikidata. It combines logical, intra-sentence, and co-reference reasoning with path-based interpretability.

D.2 LLM-based Baselines

GenRDK (Sun et al., 2024) A zero-shot document-level relation triplet extraction framework that employs a chain-of-retrieval and denoising strategy to steer LLMs in understanding relations and generating high-quality synthetic data. Their model is fine-tuned with LLaMA2-13B-CHAT.

DiVA-DocRE^{GT} (Wu et al., 2024a) A Discriminative and Voice-Aware (DiVA) paradigm for DocRE. It first identifies the relation types present in the document and then extracts subject and object entities leveraging both active and passive voice information. They report results in three distinct experimental settings: Dev^{Z}, a zero-shot setting where ChatGPT is used to generate triplets; Dev^{FT}, where LLama3-7B is fine-tuned to generate triplets; and Dev^{GT}, which uses ground-truth relations to guide triplet generation.

REPLM (Ozyurt et al., 2024) A method for in-context few-shot relation extraction leveraging pre-trained language models. For each relation in the relation list, REPLM generates candidate triplets and filters them by calculating the joint probabilities of subject-object pairs. Final predictions are obtained by aggregating the outputs from multiple retrieved in-context demonstrations.

AutoRE (Xue et al., 2024) An end-to-end model that integrates LLMs with QLoRA (Dettmers et al., 2023) under a novel relation extraction paradigm named RHF (Relation-Head-Facts). It first identifies the relation types present in the document, then extracts the corresponding head entities for each relation, and finally generates complete triplets based on the identified relations and heads.

E Analysis of Token Usage

Since the iterative reflection framework requires repeated calls to LLMs, it naturally raises concerns about increased computational cost and runtime. To quantify this, we provide detailed token usage statistics for two iterations of the 998 documents in DocRED (calculated using OpenAI's tiktoken) in Table 4.

Process		Token
Initial Extraction		3347715
Iteration 1	Reconstruct Document	893874
	Compare Two Documents	6807872
	Correct Triplet Entity Type Errors	187762
	Inference	1358232
	One-to-one Filtering	88264
	Mutual Exclusion Constraints	203190
	Directionality Constraints	46679
Iteration 2	Reconstruct Document	1010179
	Compare Two Documents	6941132
	Correct Triplet Entity Type Errors	33732
	Inference	1365856
	One-to-one Filtering	47000
	Mutual Exclusion Constraints	49056
	Directionality Constraints	8635

Table 4: Detailed token usage statistics for two iterations of DocRED dataset (calculated using OpenAI's tiktoken).

Notably, we observe decreasing token usage in later iterations for entity type correction, one-to-one filtering, mutual exclusion constraints and directionality constraints, demonstrating the framework's increasing efficiency. While iterative calls inherently require more tokens than single-pass methods, our analysis shows comparable costs to self-consistency approaches while achieving superior performance. As shown in Table 1, our method is applicable to all LLMs and using smaller open-source models (e.g., ChatGLM3-6B, LLama3-8B) can significantly reduce API costs while maintaining effectiveness.

F Case Study

Table 13 to Table 19 present the case of a complete iteration process.

Relation	Description	Subject Types	Object Types
head of government	Object is the head of government of subject. Subject can be a town, city, municipality, state, country, or other governmental body.	LOC	PER
country	Object is the sovereign state of subject. Subject is not a person.	LOC, ORG, MISC	LOC, MISC
place of birth	Object is the most specific known birth location of the subject. Subject refers to the person, animal, or fictional character. Object refers to the most specific known location of birth (e.g., a hospital, city, or even a particular building or place).	PER	LOC
place of death	Object is the most specific known death location of subject. Subject refers to the person, animal, or fictional character.Object refers to the most specific known location of death (e.g., a hospital, city, or specific place within a city).	PER	LOC
father	Object is the father of subject.	PER	PER
mother	Object is the mother of subject.	PER	PER
spouse	Subject has object as their spouse (husband, wife, partner, etc.)	PER	PER
country of citizenship	Object is a country that recognizes subject as its citizen.	PER, MISC	LOC, MISC
continent	Object is the continent of which subject is a part.	LOC, ORG, MISC	LOC
head of state	Object is the official with the highest formal authority in subject. Subject refers to the country or state where the object holds the highest formal authority. Object refers to the official (such as the president, monarch, or prime minister).	LOC, ORG	PER
position held	Subject currently or formerly holds object position or public office.	PER	ORG, MISC
child	Object is the offspring (son or daughter) of subject. Subject refers to the parent. Object refers to the offspring (son or daughter) of the subject, regardless of age.	PER	PER
member of sports team	Object is the sports team or club that subject currently or formerly represents. Subject refers to the individual (e.g., an athlete or player). Object refers to the sports team or club that the subject currently or formerly represented.	PER	ORG, LOC, MISC
educated at	Object is the educational institution attended by subject. Subject refers to the individual. Object refers to the educational institution (e.g., university, school, or college) that the subject attended.	PER	ORG, LOC
composer	Object is the person(s) who wrote the music for subject. Subject refers to the musical work (e.g., song, score, composition, etc.). Object refers to the person(s) who composed or wrote the music for the subject.	MISC	PER
member of political party	Object is the political party of which subject is or has been a member. Subject refers to the politician. Object refers to the political party that the subject is or has been a member of.	PER, ORG, MISC	ORG, LOC
employer	Object is the person or organization for which subject works or worked. Subject refers to the individual (e.g., employee, worker). Object refers to the person or organization that the subject works for or has worked for in the past.	PER	ORG, MISC, LOC

Table 5: Examples of relation descriptions and entity type constraints for DocRED and Re-DocRED datasets.

Reconstruct Extracted Triplets into a Document

Task: Given a set of relation triplets, a list of relation types with descriptions, and all entity mentions for each entity in the relation triplets, your task is to reconstruct the triplets into a coherent paragraph. The paragraph should rephrase the content of the triplets into natural language, while carefully maintaining the direction of the relations between entities. The paragraph should not contain any additional content or explanations.

Instructions:
1. For each relation triplet, use the subject, relation, and object to create a natural language sentence that describes the relationship.
2. Pay close attention to the direction of the relation. For example, if the triplet is {'sub': 'Hampshire County', 'rel': 'located in the administrative territorial entity', 'obj': 'West Virginia'}, the sentence should reflect that Hampshire County is located in West Virginia, not the other way around.
3. The paragraph should only contain these sentences and should not include any extra context or information not directly related to the triplets.
4. Ensure that the paragraph is coherent and logically structured, with proper flow between sentences.
5. The entities should be referenced by their full names as given in the triplets (e.g., "Washington Place," "William Washington House," etc.).

Triplets: *[Extracted Triplets List T]*
Relation Types and Descriptions: *[{'rel': [Relation Name R], 'description': [Relation Description RD]}]*
Entities: *[{'entity': [Entity Name E], 'entity type': [Entity Type ET]}]*
Reconstructed paragraph:

Table 6: Prompt for reconstructing extracted triplets into a document.

Compare Reconstructed Document with Original Document and Identify Missing Triplets

Given an original document, all entity mentions of each entity in the document, target relation type with description and a document reconstructed from previously extracted relation triplets, your task is to compare two documents and indicate which information about *[Relation Name R]* relation type in the original document is not in the reconstructed document. Then identify all missing triplets of *[Relation Name R]* relation type, which means *[Relation Description RD]*. You can consider whether each pair of entities in the entity list has a *[Relation Name R]* relation.

Here is an example, please output according to the format of the example.
Original document: *[Example Original Document ED]*
Reconstructed document: *[Example Reconstructed Document ED']*
Entities: *[{'entity': [Example Entity Name EE'], 'entity type': [Example Entity Type ET']}]*
Missing information: *[A Text about Missing Information MI]*
Missing triplets: *[Missing Triplets List MT]*

Now refer to the example above and compare the following two documents:
Original document: *[Original Document D]*
Reconstructed document: *[Reconstructed Document D']*
Entities: *[{'entity': [Entity Name E], 'entity type': [Entity Type ET]}]*
Missing information:

Table 7: Prompt for comparing reconstructed document with original document and identifying missing triplets.

Correct Triplet Entity Type Errors

Given a target relation type list, a document, and all entity mentions of each entity in the document, your task is to modify the triplet where either the subject or the object entity type is incorrect (the subject or object will be specified after the triplet). Please select the correct entity from the given entity list and modify the triplet accordingly. Please modify only the subject or object, or both the subject and object that entity type is incorrect. Do not change any other part of the triplet. Only output the modified triplet.

Relation Type and Description: *{'rel': [Error Triplet's Relation Name ETR], 'description': [Error Triplet's Relation Description ETRD]}*
The subject of the *[Error Triplet's Relation Name ETR]* relation can only be of type *[Subject Type List ST]*, and the object can only be of type *[Object Type List OT]*.

Document: *[Original Document D]*
Entities: *[{'entity': [Entity Name E], 'entity type': [Entity Type ET]}]*

Triplet where either subject or object, or both subject and object entity types are incorrect: *[Error Triplet and 'Subject entity type is incorrect!', 'Object entity type is incorrect!' or 'Both subject and object types are incorrect!']*
Please select the correct subject or object from the given entities list, modify the triplet and output the modified version.
The modified triplet:

Table 8: Prompt for correcting triplet entity type errors.

Inference

Given a target relation type list, a document, and all entity mentions of each entity in the document, please extract all valid given relation types between any two given entities in the document. Each line outputs an extracted relation triple, and the format of each triplet is: {'sub': subject entity, 'rel': relation type, 'obj': object entity}. Each relation triplet should be output only once.

Relation Types and Descriptions: *[{'rel': [Relation Name R], 'description': [Relation Description RD]}]*
Document: *[Original Document D]*
Entities: *[{'entity': [Entity Name E], 'entity type': [Entity Type ET]}]*
All relation triplets extracted from the document in the previous iteration will be scored and reordered based on the degree and logical rules of the subject and object entities, with higher scores given to the higher order. **Here are the reordered triplets extracted from the previous iteration:** *[Reordered Triplets List RT]*

Through logical reasoning, the following triplets may have been incorrectly extracted: *[Inferred Incorrectly Extracted Triplets List IIET]*
Based on all the relation triplets extracted in previous iterations and logical reasoning, the following triplets may still not have been extracted: *[Inferred Unextracted Triplets List IUT]*

Please refer to the above feedback and extract the triplets of the target relation types again from the original document. **All relation triplets extracted from the document:**

Table 9: Prompt for inference.

One-to-one Filtering

Given a document, all entity mentions in the document, and the relation description, your task is to select the unique correct triplet from the previously extracted triplets. Please only output the triplet you believe to be the correct one.

Document: *[Original Document D]*
Entities: *[{'entity': [Entity Name E], 'entity type': [Entity Type ET]}]*
Relation Type and Description: *{'rel': [Two Triplets' Relation Name TTR], 'description': [Two Triplets' Relation Description TTRD]}*

The previously extracted triplets: *[Two Contradictory Triplets T1 and T2]*
The unique correct triplet (Choose from the two triplets above. Do not modify the content of triplets):

Table 10: Prompt for one-to-one filtering.

Mutual Exclusion Constraints

Given a document, all entity mentions in the document, and the relation description, your task is to select the unique correct triplet from the previously extracted triplets. Please only output the triplet you believe to be the correct one.

Document: *[Original Document D]*
Entities: *[{'entity': [Entity Name E], 'entity type': [Entity Type ET]}]*
Relation Type and Description: *[{'rel': [Two Mutual Exclusion Relation Name MER], 'description': [Two Mutual Exclusion Relation Description MERD]}]*

The previously extracted triplets: *[Two Mutual Exclusion Triplets T1 and T2]*
The unique correct triplet (Choose from the two triplets above. Do not modify the content of triplets):

Table 11: Prompt for mutual exclusion constraints.

Directionality Constraints

Given a document and the relation description, your task is to select the unique correct triplet from the previously extracted triplets. Please pay attention to the direction of the relation, which is *[Two Triplets' Relation Description TTRD]* and only output the triplet you believe to be the correct one.

Document: *[Original Document D]*
Relation Type and Description: *{'rel': [Two Triplets' Relation Name TTR], 'description': [Two Triplets' Relation Description TTRD]}*

The previously extracted triplets: *[Two Contradictory Triplets T1 and T2]*
The unique correct triplet (Choose from the two triplets above. Do not modify the content of triplets):

Table 12: Prompt for directionality constraints.

Initial Extraction
F1: 0.235 **Precision: 0.385** **Recall: 0.169**
{'sub': 'VivoTab RT LTE', 'rel': 'follows', 'obj': 'VivoTab RT 3 G'} ×
{'sub': 'VivoTab RT 3 G', 'rel': 'follows', 'obj': 'VivoTab RT LTE'} ×
{'sub': 'VivoTab RT 3 G', 'rel': 'followed by', 'obj': 'VivoTab RT LTE'} ×
{'sub': 'VivoTab RT', 'rel': 'followed by', 'obj': 'VivoTab RT 3 G'} ✓
{'sub': 'VivoTab', 'rel': 'series', 'obj': 'Vivo'} ✓
{'sub': 'VivoTab', 'rel': 'subclass of', 'obj': 'Vivo'} ✓
{'sub': 'VivoTab', 'rel': 'has part', 'obj': 'VivoTab RT 3 G'} ✓
{'sub': 'VivoTab Note 8', 'rel': 'follows', 'obj': 'VivoTab Smart'} ×
{'sub': 'VivoTab Smart', 'rel': 'follows', 'obj': 'VivoTab Note 8'} ×
{'sub': 'VivoTab', 'rel': 'follows', 'obj': 'VivoTab RT'} ×
{'sub': 'VivoTab RT LTE', 'rel': 'follows', 'obj': 'VivoTab Smart'} ×
{'sub': 'Windows 8.1', 'rel': 'follows', 'obj': 'Windows 8'} ✓
{'sub': 'VivoTab RT', 'rel': 'followed by', 'obj': 'VivoTab Note 8'} ×
{'sub': 'VivoTab', 'rel': 'followed by', 'obj': 'VivoTab Note 8'} ×
{'sub': 'VivoTab Smart', 'rel': 'followed by', 'obj': 'VivoTab Note 8'} ×
{'sub': 'VivoTab RT LTE', 'rel': 'followed by', 'obj': 'VivoTab Smart'} ×
{'sub': 'VivoTab', 'rel': 'manufacturer', 'obj': 'Asus'} ✓
{'sub': 'VivoTab', 'rel': 'developer', 'obj': 'Asus'} ✓
{'sub': 'VivoTab', 'rel': 'has part', 'obj': 'VivoTab Note 8'} ×
{'sub': 'VivoTab', 'rel': 'has part', 'obj': 'VivoTab RT LTE'} ✓
{'sub': 'VivoTab', 'rel': 'has part', 'obj': 'VivoTab Smart'} ✓
{'sub': 'VivoTab', 'rel': 'has part', 'obj': 'VivoTab RT'} ✓
{'sub': 'VivoTab', 'rel': 'has part', 'obj': 'Asus WebStorage'} ×
{'sub': 'VivoTab', 'rel': 'has part', 'obj': 'Windows 8.1'} ×
{'sub': 'VivoTab', 'rel': 'has part', 'obj': 'Windows 8'} ×
{'sub': 'Asus', 'rel': 'product or material produced', 'obj': 'VivoTab'} ×

Table 13: Initial Extraction.

Reconstructed Document
The VivoTab RT LTE follows the VivoTab RT 3 G, while the VivoTab RT 3 G is followed by the VivoTab RT LTE. The VivoTab RT is followed by the VivoTab RT 3 G, and the VivoTab RT LTE is followed by the VivoTab Smart. The VivoTab Note 8 follows the VivoTab Smart, and the VivoTab Smart is followed by the VivoTab Note 8. The VivoTab follows the VivoTab RT and is followed by the VivoTab Note 8. The VivoTab is part of the Vivo series and is a subclass of Vivo. Asus is both the manufacturer and developer of the VivoTab, which has several components including the VivoTab RT 3 G, VivoTab Note 8, VivoTab RT LTE, VivoTab Smart, VivoTab RT, Asus WebStorage, Windows 8.1, and Windows 8. Additionally, Asus produces the VivoTab. Windows 8.1 follows Windows 8.
Original Document
VivoTab is a series of Microsoft Windows hybrid tablet computers designed by Asus . It is a sub - series of the Vivo series by Asus . The name is derived from the Latin word ẗo live ände , along with Asus 's Transformer series of convertible devices running Windows , is a primary competitor to the Microsoft Surface . The family is made up of the VivoTab , VivoTab RT , VivoTab RT 3 G , VivoTab RT LTE , VivoTab Smart , and later on the VivoTab Note 8 . All of the tablets come with Windows 8 (or Windows 8.1 on the Note 8) , a 3-year subscription to Asus WebStorage . They have high definition screens advertise ultra - portability and extended battery life , and the ability detachable tablets . VivoTab RT has an MSRP of $599USD(32GB)and$ 699 (64 GB)

Table 14: Reconstructed document and original document. The content highlighted in red in the reconstructed document does not align with the original document and corresponds to incorrect triplets. Meanwhile, the content highlighted in red in the original document does not appear in the reconstructed document, representing triplets that have not been extracted.

Comparsion		
F1: 0.432	**Precision: 0.462**	**Recall: 0.407**
Add:		
{'sub': 'VivoTab RT LTE', 'rel': 'follows', 'obj': 'VivoTab RT'} ×		
{'sub': 'VivoTab RT', 'rel': 'series', 'obj': 'VivoTab'} ✓+		
{'sub': 'VivoTab RT 3 G', 'rel': 'series', 'obj': 'VivoTab'} ✓+		
{'sub': 'VivoTab RT LTE', 'rel': 'series', 'obj': 'VivoTab'} ✓+		
{'sub': 'VivoTab Smart', 'rel': 'series', 'obj': 'VivoTab'} ✓+		
{'sub': 'VivoTab Note 8', 'rel': 'series', 'obj': 'VivoTab'} ×		
{'sub': 'VivoTab RT', 'rel': 'manufacturer', 'obj': 'Asus'} ✓+		
{'sub': 'VivoTab RT 3 G', 'rel': 'manufacturer', 'obj': 'Asus'} ✓+		
{'sub': 'VivoTab RT LTE', 'rel': 'manufacturer', 'obj': 'Asus'} ✓+		
{'sub': 'VivoTab Smart', 'rel': 'manufacturer', 'obj': 'Asus'} ✓+		
{'sub': 'VivoTab Note 8', 'rel': 'manufacturer', 'obj': 'Asus'} ✓+		
{'sub': 'VivoTab RT', 'rel': 'developer', 'obj': 'Asus'} ✓+		
{'sub': 'VivoTab RT 3 G', 'rel': 'developer', 'obj': 'Asus'} ✓+		
{'sub': 'VivoTab RT LTE', 'rel': 'developer', 'obj': 'Asus'} ✓+		
{'sub': 'VivoTab Smart', 'rel': 'developer', 'obj': 'Asus'} ✓+		
{'sub': 'VivoTab Note 8', 'rel': 'developer', 'obj': 'Asus'} ✓+		
{'sub': 'Asus', 'rel': 'product or material produced', 'obj': 'VivoTab RT'} ×		
{'sub': 'Asus', 'rel': 'product or material produced', 'obj': 'VivoTab RT 3 G'} ×		
{'sub': 'Asus', 'rel': 'product or material produced', 'obj': 'VivoTab RT LTE'} ×		
{'sub': 'Asus', 'rel': 'product or material produced', 'obj': 'VivoTab Smart'} ×		
{'sub': 'Asus', 'rel': 'product or material produced', 'obj': 'VivoTab Note 8'} ×		
{'sub': 'VivoTab RT', 'rel': 'subclass of', 'obj': 'VivoTab'} ×		
{'sub': 'VivoTab RT 3 G', 'rel': 'subclass of', 'obj': 'VivoTab'} ×		
{'sub': 'VivoTab RT LTE', 'rel': 'subclass of', 'obj': 'VivoTab'} ×		
{'sub': 'VivoTab Smart', 'rel': 'subclass of', 'obj': 'VivoTab'} ×		
{'sub': 'VivoTab Note 8', 'rel': 'subclass of', 'obj': 'VivoTab'} ×		

Table 15: Compare reconstructed document and original document to identify unextracted triplets.

Inference		
F1: 0.456	**Precision: 0.473**	**Recall: 0.441**
Add:		
{'sub': 'Windows 8.1', 'rel': 'follows', 'obj': 'Windows 8'} ⇒		
{'sub': 'Windows 8', 'rel': 'followed by', 'obj': 'Windows 8.1'} ✓+		
{'sub': 'VivoTab RT', 'rel': 'followed by', 'obj': 'VivoTab RT 3 G'} ⇒		
{'sub': 'VivoTab RT 3 G', 'rel': 'follows', 'obj': 'VivoTab RT'} ✓+		
{'sub': 'VivoTab RT LTE', 'rel': 'followed by', 'obj': 'VivoTab Smart'} ⇒		
{'sub': 'VivoTab Smart', 'rel': 'follows', 'obj': 'VivoTab RT LTE'} ×		

Table 16: Inference based on logical rules to identify unextracted triplets.

One-to-one		
F1: 0.460	**Precision: 0.481**	**Recall: 0.441**
{'sub': 'VivoTab RT', 'rel': 'followed by', 'obj': 'VivoTab RT 3 G'} ✓		
{'sub': 'VivoTab RT', 'rel': 'followed by', 'obj': 'VivoTab Note 8'} × **(Delete)**		

Table 17: Delete incorrect triplets based on one-to-one filtering.

Mutual Exclusion Constraint
F1: 0.491 **Precision: 0.553** **Recall: 0.441**
{'sub': 'VivoTab', 'rel': 'has part', 'obj': 'VivoTab RT'} ✓
{'sub': 'VivoTab', 'rel': 'follows', 'obj': 'VivoTab RT'} × **(Delete)**
{'sub': 'VivoTab', 'rel': 'has part', 'obj': 'VivoTab Note 8'} ×
{'sub': 'VivoTab', 'rel': 'followed by', 'obj': 'VivoTab Note 8'} × **(Delete)**
{'sub': 'VivoTab RT', 'rel': 'series', 'obj': 'VivoTab'} ✓
{'sub': 'VivoTab RT', 'rel': 'subclass of', 'obj': 'VivoTab'} × **(Delete)**
{'sub': 'VivoTab RT 3 G', 'rel': 'series', 'obj': 'VivoTab'} ✓
{'sub': 'VivoTab RT 3 G', 'rel': 'subclass of', 'obj': 'VivoTab'} × **(Delete)**
{'sub': 'VivoTab RT LTE', 'rel': 'series', 'obj': 'VivoTab'} ✓
{'sub': 'VivoTab RT LTE', 'rel': 'subclass of', 'obj': 'VivoTab'} × **(Delete)**
{'sub': 'VivoTab Smart', 'rel': 'series', 'obj': 'VivoTab'} ✓
{'sub': 'VivoTab Smart', 'rel': 'subclass of', 'obj': 'VivoTab'} × **(Delete)**
{'sub': 'VivoTab Note 8', 'rel': 'series', 'obj': 'VivoTab'} ×
{'sub': 'VivoTab Note 8', 'rel': 'subclass of', 'obj': 'VivoTab'} × **(Delete)**

Table 18: Delete incorrect triplets based on mutual exclusion constraints.

Directionality Constraint
F1: 0.505 **Precision: 0.591** **Recall: 0.441**
{'sub': 'VivoTab RT 3 G', 'rel': 'follows', 'obj': 'VivoTab RT LTE'} × **(Delete)**
{'sub': 'VivoTab RT LTE', 'rel': 'follows', 'obj': 'VivoTab RT 3 G'} ×
{'sub': 'VivoTab RT LTE', 'rel': 'follows', 'obj': 'VivoTab Smart'} × **(Delete)**
{'sub': 'VivoTab Smart', 'rel': 'follows', 'obj': 'VivoTab RT LTE'} ×
{'sub': 'VivoTab Note 8', 'rel': 'follows', 'obj': 'VivoTab Smart'} ×
{'sub': 'VivoTab Smart', 'rel': 'follows', 'obj': 'VivoTab Note 8'} × **(Delete)**

Table 19: Delete incorrect triplets based on directionality constraints.