VERIFYING JAPANESE DOCUMENT-LEVEL RELATION TRIPLES WITH STRUCTURED OUTPUTS AND TYPE CONSTRAINTS

Anonymous authors
Paper under double-blind review

ABSTRACT

Low-cost large language models (LLMs) are appealing for Japanese document-level relation extraction (DocRE) and knowledge graph construction, but one-shot prompting often over-generates relation triples, producing many false positives and impractically low precision. This is difficult because DocRE requires cross-sentence context and because Japanese discourse and limited resources increase ambiguity, making it easy for a generative model to output plausible but unsupported relations. We address this by separating "find candidates" from "decide correctness" in a Two-Stage pipeline: a recall-oriented candidate generator followed by a precision-oriented verifier, with both stages forced to emit schema-valid JSON via structured, JSON Schema–constrained decoding. We then apply deterministic relation-specific domain and range filtering using type-pair constraints learned from the training data. On 10 JacRED development documents selected via character-length–based stratified sampling, the pipeline consistently reduces false positives across Gemini Flash configurations (2.0/2.5/3 preview; with and without thinking), yielding large precision gains (up to +0.18 absolute) and improving best F1 from 0.17–0.26 (baseline) to 0.25–0.27 (two-stage) depending on configuration. These results suggest that verification plus simple, data-driven type constraints can make low-cost LLM DocRE substantially more reliable without training task-specific models.

1 INTRODUCTION

Document-level relation extraction (DocRE) seeks to identify semantic relations between entity pairs using evidence that may be distributed across multiple sentences, sometimes requiring multi-hop reasoning and careful handling of coreference and entity aliasing. Benchmarks such as DocRED established that a large fraction of relations cannot be recovered from a single sentence and emphasized the computational and modeling challenges arising from quadratic numbers of entity pairs and cross-sentence evidence patterns Yao et al. (2019); Delaunay et al. (2023). While supervised DocRE architectures have progressed considerably, the rise of large language models (LLMs) has renewed interest in prompt-based extraction for settings where model training is undesirable or infeasible. However, surveys of LLM-based generative information extraction repeatedly highlight a tension between flexibility and reliability: LLMs can follow natural language instructions, but they can also hallucinate, drift from requested formats, and behave unpredictably under distribution shift or minor prompt changes Xu et al. (2023).

These reliability issues are particularly salient for Japanese DocRE. JacRED, the first public Japanese Wikipedia-based DocRE benchmark, was created to fill a major resource gap and to quantify the limitations of cross-lingual transfer from English DocRE data Ma et al. (2024). The dataset construction study reports that topic and surface-structure mismatches make translation-based transfer insufficient for strong performance on native Japanese documents, and that in-context LLM baselines are weak under typical prompting regimes Ma et al. (2024). Despite these limitations, low-cost LLM endpoints remain attractive in practice because they reduce engineering overhead and enable rapid prototyping for knowledge graph construction pipelines.

In such practical pipelines, the most straightforward approach is one-shot extraction: a single LLM call that requests all relation triples in a document. Our experiments show a recurring failure mode under this setting: the model produces a large number of false positive triples, leading to low precision and high downstream cost for manual validation or graph cleanup. This motivates the central question of this paper: how can we make low-cost, training-free LLM-based Japanese DocRE more precise without sacrificing the simplicity that makes these tools appealing?

We explore a minimal reliability strategy grounded in a simple decomposition: open-ended extraction and correctness judgment are different cognitive tasks, and the latter can be easier for LLMs. We therefore separate "candidate generation" from "verification," and we additionally enforce strict output structure so that intermediate artifacts can be handled deterministically. Specifically, we propose a Two-Stage pipeline in which Stage 1 generates a recall-oriented list of candidate triples, Stage 2 verifies candidates in batches to prune unsupported relations, and a final deterministic filter removes triples that violate relation-specific domain and range patterns observed in the training data. Both stages use JSON Schema–constrained decoding ("Structured Outputs"), aligning with recent evidence that constrained decoding improves schema compliance and can improve downstream task quality by eliminating malformed outputs and reducing ambiguity at the interface between models and programs Geng et al. (2025).

We evaluate this approach on 10 documents from the JacRED development set selected via character-length–based stratified sampling. We compare a Baseline one-shot extraction method to the Proposed Two-Stage pipeline across multiple Gemini Flash family configurations: Gemini 2.0 Flash, Gemini 2.5 Flash, and Gemini 3 Flash Preview, each evaluated with thinking either disabled (none/off) or enabled at a budget of 2048. We report micro-averaged precision, recall, and F1 over the 10 documents, along with the underlying counts of true positives (TP), false positives (FP), and false negatives (FN).

Across all tested configurations, the proposed pipeline reduces false positives and increases precision, with improvements ranging from +0.06 to +0.18 absolute precision depending on configuration. The best observed F1 reaches 0.27, achieved by Gemini 2.5 Flash with thinking enabled (2048) and by Gemini 3 Flash Preview with thinking off. At the same time, overall recall remains low (0.12–0.22), indicating that candidate generation and entity alignment, rather than verification capacity, are the primary bottlenecks.

Our contributions are: - We demonstrate that a verification-first decomposition for Japanese DocRE can systematically reduce false positives for low-cost LLM extraction, improving precision and, in several configurations, F1. - We show that JSON Schema–constrained structured outputs are a practical interface for building robust multi-call DocRE pipelines, enabling deterministic parsing in both candidate generation and verification. - We introduce and evaluate a simple, deterministic relation-wise type-pair filter derived from training data that further suppresses semantically implausible triples without additional model calls.

The experiments also reveal clear remaining challenges. In particular, low recall suggests that future work should focus on better candidate generation policies and on robustness to entity alignment and relation directionality errors. More broadly, the paper suggests that reliability improvements for LLM-based DocRE can come not only from larger models or additional training, but also from careful task decomposition and lightweight constraints that exploit dataset structure.

2 Related Work

We position our work at the intersection of document-level relation extraction, Japanese DocRE resources, and reliability techniques for LLM-based generative information extraction. Across these areas, prior work provides strong motivation and relevant tools, but it also leaves a gap for cost-sensitive, training-free, precision-oriented extraction pipelines.

DocRE benchmarks and supervised modeling. DocRED established the now-standard document-level formulation of relation extraction over Wikipedia documents linked to a knowledge base, emphasizing that many relations require cross-sentence evidence and introducing evaluation practices such as micro-F1 and overlap-robust "Ign" variants Yao et al. (2019). Subsequent work has produced a wide range of supervised DocRE models, including graph-based and transformer-based designs that explicitly address cross-sentence reasoning and entity-centric aggregation. The DocRE survey by Delaunay

et al. synthesizes these approaches and highlights persistent challenges such as computational cost, candidate pair explosion, and sensitivity to upstream coreference/NER pipelines Delaunay et al. (2023). In contrast, our work does not attempt to compete with supervised DocRE accuracy; instead, we study how to make low-cost, prompt-based LLM extraction more usable when training and heavy model development are not options.

Japanese DocRE resources and transfer limitations. JacRED was introduced to address the lack of general-purpose Japanese DocRE datasets and includes evidence sentences, 8 entity types, and 35 relations derived from a reduced and augmented schema Ma et al. (2024). The JacRED study also reports that translation-based cross-lingual transfer yields low recall on native Japanese documents and that LLM in-context learning baselines perform poorly, underscoring a practical gap between available resources and reliable extraction in Japanese Ma et al. (2024). Our work leverages JacRED as an evaluation setting but reframes the objective: rather than improving supervised model training, we aim to reduce false positives in a lightweight LLM pipeline.

LLM-based generative IE and reliability mechanisms. LLMs have been widely applied to information extraction via prompting, in-context learning, and generative formulations, but survey evidence emphasizes brittleness, hallucinations, and sensitivity to prompt choices, particularly in low-resource or zero-/few-shot regimes Xu et al. (2023). Many approaches in this space pursue accuracy through fine-tuning, universal extraction frameworks, or sophisticated self-improvement loops. Our perspective differs primarily in assumptions: we restrict ourselves to low-cost endpoints and avoid training, so we emphasize post-hoc reliability controls that can be implemented purely through prompting, constrained generation, and deterministic filtering.

Structured outputs and constrained decoding. Constrained decoding for JSON Schema and related grammars has been studied as a way to force well-formed outputs, with recent benchmark work demonstrating meaningful differences in schema coverage and compliance across engines and showing that structured outputs can sometimes improve downstream task accuracy by improving the model-program interface Geng et al. (2025). Our work adopts structured outputs not as an end in itself, but as a mechanism that enables decomposition (generate then verify) and deterministic post-processing in DocRE. The novelty is therefore not the constraint technology, but the use of strict schemas to make a two-stage DocRE pipeline reliable and inexpensive.

Joint extraction models. Joint NER+RE models reduce pipeline error propagation by learning entities and relations together, for example via BERT-based tagging coupled with biaffine relation classification Giorgi et al. (2019). While conceptually related to our goal of improving end-to-end extraction quality, these models are designed for supervised sentence-level settings and depend on token-level annotations. They are therefore not directly applicable to our document-level, LLM-only, no-training setting, where the main challenge is suppressing false positives from generative extraction rather than optimizing supervised classification accuracy.

3 BACKGROUND

DocRE task definition and assumptions. We follow the standard DocRE framing in which the input is a document D and the output is a set of directed relation instances between entity pairs. A document contains multiple sentences and references a set of entities E; each entity can have multiple mentions in the document and is associated with a coarse entity type Yao et al. (2019). For JacRED, the entity type inventory comprises 8 IREX-style categories, and the relation schema contains 35 directed relations (including inverse relations added after annotation) Ma et al. (2024). A predicted relation instance is represented as a triple (h, r, t), where h is the head entity, t is the tail entity, and r is a relation label.

Dataset characteristics. JacRED is a Japanese Wikipedia-based DocRE benchmark designed to provide a native Japanese evaluation setting and to expose limits of cross-lingual transfer from English DocRE datasets Ma et al. (2024). It contains 2,000 documents and 42,241 annotated triples, split into train/dev/test. In this paper, JacRED provides the document text, entity types, and gold relation triples used for evaluation.

Evaluation protocol and metrics. We evaluate extraction as set prediction over relation triples. Let G(D) be the gold set of triples for document D and P(D) be the predicted set. Across a collection of documents, we compute micro-averaged counts of true positives TP = $|P \cap G|$, false positives FP

= |P G|, and false negatives FN = |G P|, and then compute Precision = TP / (TP + FP), Recall = TP / (TP + FN), and F1 = 2PR / (P + R). While DocRED also introduced "Ign" metrics to reduce the effect of train-test overlap in supervised settings Yao et al. (2019), our evaluation uses standard micro-averaged precision/recall/F1 over the selected documents.

Structured outputs for generative extraction. In an LLM-based generative approach, DocRE can be treated as conditional generation in which the model is prompted with a document and asked to emit extracted relations. A recurring practical challenge is output controllability: the model may produce outputs that are hard to parse or that mix analysis with extraction. Schema-constrained decoding addresses this by restricting the model's output token-by-token so that it must form a valid instance of a target JSON Schema, improving syntactic compliance and enabling deterministic parsing Geng et al. (2025).

Type-pair constraints as a deterministic prior. Many relation schemas implicitly constrain argument types through domain and range expectations. With discrete entity types, we can represent this as a relation-specific set of allowed type pairs. For each relation r, define C(r) as the set of (type(h), type(t)) pairs that occur with relation r in the training data. Given a predicted triple (h, r, t), a deterministic filter retains the triple only if (type(h), type(t)) is in C(r). This mechanism does not require additional learning or model inference; it is a simple data-derived constraint that can reduce implausible predictions.

Problem setting. The paper studies low-cost, prompt-based Japanese DocRE for knowledge graph triple extraction. The input is a Japanese document from the JacRED development set; the output is a set of relation triples over the JacRED schema. We compare a one-shot extraction baseline against a two-stage extract-then-verify pipeline, both implemented with schema-constrained structured outputs, and we evaluate them using micro-averaged precision, recall, and F1 over 10 sampled documents.

4 METHOD

We propose a reliability-oriented extraction framework designed for low-cost LLM endpoints where the main failure mode is over-generation: the model returns many plausible-looking relation triples that are not supported by the document. The core idea is to decompose the interaction into two simpler subproblems that better match the LLM's capabilities and to make every intermediate artifact machine-checkable. First, we ask the model to enumerate candidate triples with a bias toward recall. Second, we ask the model to verify each candidate against the same document, turning open-ended extraction into a more constrained decision task. Finally, we apply a deterministic type-compatibility filter derived from the training data to remove remaining implausible outputs.

Stage 1: recall-oriented candidate generation. Given a document D, the system prompts the LLM to produce a list of candidate relation triples. The generation is constrained to a fixed JSON structure via an extraction schema (EXTRACTION_SCHEMA). This design serves two purposes. It encourages broad coverage (so that later verification has something to prune) while ensuring that the output can be parsed deterministically without heuristic cleanup. The use of schema-constrained decoding follows the general constrained generation paradigm described in structured-output studies Geng et al. (2025).

Stage 2: precision-oriented batch verification. Stage 1 can over-generate; Stage 2 addresses this by verifying candidates. Each candidate triple (h, r, t) is presented to the model along with the document, and the model is asked to judge whether the relation is supported. Verification outputs are also constrained to a schema (VERIFICATION_SCHEMA), which forces an unambiguous structured decision per candidate. To control API cost, candidates are verified in batches of size 10, allowing a single call to handle multiple decisions. This stage targets the specific one-shot failure mode: hallucinated or weakly supported relations should be easier to reject in a verification setting than to avoid generating in the first place.

Deterministic post-processing with type-pair constraints. Even after verification, some false positives can persist, particularly those that are linguistically plausible but schema-inconsistent given entity types. We therefore apply relation-wise type constraints computed from the training data. For each relation r, we precompute the allowed set C(r) of observed (head_type, tail_type) pairs. For each verified triple (h, r, t), we look up type(h) and type(t) and retain the triple only if (type(h), type(t)) is

in C(r). This final step is deterministic, adds no inference cost, and explicitly leverages the dataset's type system as a lightweight prior.

Baseline for comparison. The baseline system uses one-shot extraction: a single LLM call is asked to extract entities and relations simultaneously and return a set of triples, followed by simple filtering. It does not include a verification step and does not apply the training-data type-pair constraint filter.

End-to-end procedure. The complete pipeline is: (1) generate candidates via Stage 1 structured output; (2) verify candidates in Stage 2 using batched structured output decisions; (3) discard candidates judged incorrect; (4) apply C(r) type-pair filtering; (5) output the remaining triples for evaluation.

5 Experimental Setup

Evaluation dataset and document selection. We evaluate on JacRED, a Japanese Wikipedia-based DocRE dataset with 2,000 documents, 35 relations, and 8 entity types Ma et al. (2024). To keep the experiment low-cost and manually manageable, we restrict evaluation to 10 documents from the JacRED development split. These documents are selected using character-length–based stratified sampling, intended to cover a range of document lengths rather than focusing only on short or long examples.

Experimental conditions: what is compared, varied, and controlled. We compare two extraction pipelines: (1) Baseline one-shot extraction and (2) Proposed Two-Stage extraction (candidate generation, batch verification, then type-pair filtering). We vary the underlying Gemini Flash model configuration while keeping the extraction condition fixed within each model run. The tested model configurations are Gemini 2.0 Flash (thinking recorded as none), Gemini 2.5 Flash (thinking off; thinking budget 2048), and Gemini 3 Flash Preview (thinking off; thinking budget 2048). Within each model and thinking setting, both Baseline and Proposed pipelines use the same model endpoint; the only difference is the pipeline structure and post-processing.

Implementation details that affect interpretation. The pipeline is implemented via the Gemini API using the Google GenAI SDK. Both stages use Structured Outputs (JSON Schema–constrained decoding) with two schemas: EXTRACTION_SCHEMA for candidate generation and VERIFICATION_SCHEMA for verification. Verification is performed in batches of size 10, which affects cost and may affect decision behavior by changing how many candidates are jointly evaluated in a single call. After verification, a deterministic filter removes triples violating relation-specific (head_type, tail_type) constraints computed from the training data.

Metrics and reporting. We compute micro-averaged Precision, Recall, and F1 over the 10 evaluated documents, and we report TP, FP, and FN counts from the evaluation logs. Given the small sample size (10 documents), we do not report confidence intervals or significance tests. The goal is to test a directional hypothesis about false positive suppression under a fixed low-cost regime, rather than to provide a definitive benchmark comparison.

Comparability to supervised baselines. JacRED reports supervised DocRE results under standard training regimes Ma et al. (2024), but we do not include those systems as baselines because our setting explicitly avoids training and instead evaluates practical prompt-based extraction pipelines under cost constraints.

6 Results

We report micro-averaged precision, recall, and F1 over 10 JacRED development documents for each Gemini configuration, comparing the Baseline one-shot extraction pipeline against the Proposed Two-Stage pipeline (candidate generation, batch verification, and type-pair filtering). Across all configurations, the most consistent empirical pattern is that the Proposed pipeline substantially reduces false positives, yielding higher precision. Recall remains low overall, and the effect on recall varies by configuration.

Overall trend: systematic false positive reduction. In every model and thinking setting, the Proposed pipeline produces fewer false positives than the Baseline. The absolute FP reductions are large: for example, Gemini 2.5 Flash with thinking off drops from 115 FP (Baseline) to 42 FP (Proposed), and

Gemini 2.0 Flash drops from 86 to 51. This supports the paper's hypothesis that verification plus deterministic constraints can suppress over-generation.

Effect on precision and F1 across configurations. Precision increases in all settings, with absolute gains between +0.06 and +0.18. F1 improves in some settings but not all, reflecting a precision–recall trade-off.

- Gemini 2.0 Flash (thinking: none): Baseline precision 0.20, recall 0.15, F1 0.17 (TP=22, FP=86, FN=126). Proposed precision 0.35, recall 0.19, F1 0.25 (TP=28, FP=51, FN=121). Observation: the pipeline both reduces FP and increases TP, yielding a clear F1 gain.

- Gemini 2.5 Flash (thinking: off): Baseline precision 0.17, recall 0.16, F1 0.17 (TP=24, FP=115, FN=124). Proposed precision 0.30, recall 0.12, F1 0.17 (TP=18, FP=42, FN=130). Observation: precision rises sharply due to FP reduction, but TP decreases, leaving F1 unchanged.

- Gemini 2.5 Flash (thinking: 2048): Baseline precision 0.18, recall 0.17, F1 0.17 (TP=25, FP=115, FN=123). Proposed precision 0.36, recall 0.21, F1 0.27 (TP=31, FP=54, FN=117). Observation: this is the best outcome for Gemini 2.5 Flash, improving both precision and recall, and achieving the joint-best F1 of 0.27.

- Gemini 3 Flash Preview (thinking: off): Baseline precision 0.26, recall 0.16, F1 0.20 (TP=24, FP=70, FN=124). Proposed precision 0.36, recall 0.22, F1 0.27 (TP=32, FP=56, FN=116). Observation: the Proposed pipeline improves all three metrics and attains F1 0.27. The accompanying analysis report highlights this configuration as a strong cost–performance point due to short runtime and strong F1.

- Gemini 3 Flash Preview (thinking: 2048): Baseline precision 0.31, recall 0.22, F1 0.26 (TP=33, FP=74, FN=115). Proposed precision 0.37, recall 0.20, F1 0.26 (TP=30, FP=52, FN=118). Observation: precision improves and FP decreases, but TP drops, leaving F1 unchanged.

Interpretations and limitations grounded in the analysis report. The analysis report attributes the consistent precision gains to the reduction of false positives achieved by verification and type constraints. It also notes that recall remains the primary bottleneck (0.12–0.22 across all systems), suggesting that failures in candidate generation and entity alignment dominate remaining error. Additionally, enabling thinking improves Baseline performance but can "over-regularize" verification in the Proposed pipeline, reducing true positives in some settings; this indicates that verification prompts and decision behavior may require calibration across inference modes.

Figures. No figures were produced or saved in the experimental logs; therefore, results are presented as numeric comparisons only.

7 Conclusion

This paper examined a practical reliability problem in low-cost Japanese document-level relation extraction: one-shot LLM prompting tends to over-generate triples, producing many false positives that undermine knowledge graph construction. We evaluated a Two-Stage approach that separates candidate generation from correctness verification, enforces schema-valid structured outputs in both stages, and applies deterministic relation-wise type-pair constraints derived from training data.

On 10 stratified JacRED development documents, the proposed pipeline consistently reduced false positives across all tested Gemini Flash configurations, yielding clear and robust precision improvements. In several configurations, these precision gains translated into higher F1, with the best observed F1 reaching 0.27. At the same time, recall remained low across both baseline and proposed pipelines, indicating that recovering missing gold relations is the dominant unresolved challenge.

Conceptually, the results support the idea that reliability for LLM-based DocRE can be improved without training by combining task decomposition, structured generation interfaces, and lightweight constraints that exploit dataset structure. Future work should focus on improving candidate generation and entity alignment, addressing relation directionality errors, and refining verification behavior to avoid unnecessary rejection of true relations under different inference modes.

This work was generated by AIRAS (Tanaka et al., 2025).

REFERENCES

Julien Delaunay, Hanh Thi Hong Tran, Carlos-Emiliano González-Gallardo, Georgeta Bordea, Nicolas Sidere, and Antoine Doucet. A comprehensive survey of document-level relation extraction. 2023.

Saibo Geng, Hudson Cooper, Michał Moskal, Samuel Jenkins, Julian Berman, Nathan Ranchin, Robert West, Eric Horvitz, and Harsha Nori. Generating structured outputs from language models: Benchmark and studies. 2025.

John Giorgi, Xindi Wang, Nicola Sahar, Won Young Shin, Gary D. Bader, and Bo Wang. End-to-end named entity recognition and relation extraction using pre-trained language models. 2019.

Youmi Ma, An Wang, and Naoaki Okazaki. Building a japanese document-level relation extraction dataset assisted by cross-lingual transfer. 2024.

Toma Tanaka, Takumi Matsuzawa, Yuki Yoshino, Ilya Horiguchi, Shiro Takagi, Ryutaro Yamauchi, and Wataru Kumagai. AIRAS, 2025. URL `https://github.com/airas-org/airas`.

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

Yuan Yao, Deming Ye, Peng Li, Xu Han, Yankai Lin, Zhenghao Liu, Zhiyuan Liu, Lixin Huang, Jie Zhou, and Maosong Sun. Docred: A large-scale document-level relation extraction dataset. 2019.