VERIFIED STRUCTURED EXTRACTION FOR JAPANESE DOCUMENT-LEVEL RELATION EXTRACTION WITH LOW-COST LLMS

Anonymous authors
Paper under double-blind review

ABSTRACT

Japanese document-level relation extraction (DocRE) is an important step for building knowledge graphs from long-form text, but one-shot prompting with low-cost large language models (LLMs) often over-generates plausible yet unsupported relation triples, yielding low precision and brittle outputs.

We propose a practical two-stage pipeline that separates (i) recall-oriented candidate proposal from (ii) precision-oriented verification, while using schema-constrained structured outputs to ensure machine-readable predictions.

On 10 stratified-sampled JacRED development documents, the pipeline consistently reduces false positives across Gemini Flash configurations and improves micro-precision in every setting, with micro-F1 improving in most configurations (up to +0.10 absolute over the one-shot baseline).

These results indicate that pairing low-cost LLM extraction with explicit verification can make Japanese DocRE outputs substantially more reliable for downstream knowledge graph construction; conceptually, our contribution is to frame precision control as a lightweight, deployable verification-and-filtering layer on top of generic LLM extraction.

1 INTRODUCTION

Document-level relation extraction (DocRE) identifies typed, directed relations between entities mentioned anywhere in a document, including cases where evidence is distributed across sentences. This capability is central to knowledge graph construction and to downstream tasks such as question answering, where relations may require aggregating clues across discourse Yao et al. (2019); Delaunay et al. (2023).

Despite strong progress in supervised DocRE, many real deployments face a different constraint: practitioners want extraction that is inexpensive, training-free, and easy to integrate. In this regime, low-cost large language models (LLMs) are attractive because they can be prompted to produce triples directly. However, document scope changes the error profile: documents contain many entities (and thus many plausible head–tail pairs), relations may be implicit, and one-shot prompting encourages models to "fill in" missing links. Our central gap is therefore practical rather than architectural: under low-cost, one-shot prompting, Japanese DocRE tends to suffer from systematically low precision due to over-generation.

This issue is amplified for Japanese. JacRED, a Japanese Wikipedia-based DocRE benchmark with evidence sentences, shows that LLM in-context learning trails supervised Japanese models and highlights language-specific phenomena (e.g., argument omission) that make relation grounding difficult Ma et al. (2024). In addition to semantic mistakes, unconstrained generation also produces formatting and parsing failures, which complicate evaluation and downstream use. Taken together, these limitations suggest that simply "prompting harder" is unlikely to yield reliable document-level triples under tight budget constraints.

We pursue a complementary direction: instead of asking a low-cost LLM to be both comprehensive and correct in one pass, we decompose extraction into proposal and verification. The proposal step is allowed to be generous (favoring coverage), while the verification step is optimized for precision

by judging whether each proposed triple is actually supported by the document. To make this decomposition practical, we use JSON Schema–constrained structured outputs to eliminate format errors and enable deterministic post-processing.

We evaluate this approach on 10 JacRED development documents selected via character-length stratified sampling and compare against a one-shot baseline across multiple Gemini Flash configurations (Gemini 2.0 Flash, Gemini 2.5 Flash, and Gemini 3 Flash Preview), including settings with and without a "thinking" budget. Across configurations, the two-stage approach consistently reduces false positives and improves precision, and improves micro-F1 in most cases, while revealing that recall remains the primary bottleneck.

Contributions: - We show that a proposal–verification decomposition materially improves the precision of low-cost LLM-based Japanese DocRE under a controlled evaluation regime. - We demonstrate that schema-constrained structured outputs enable robust, parseable relation extraction and scalable verification without custom model training. - We provide an empirical cross-configuration analysis on JacRED (10 dev documents) that characterizes when verification improves F1 and when it trades off recall, clarifying the remaining obstacles to higher end-to-end performance.

2 Related Work

Our work relates to three themes: (i) document-level relation extraction (DocRE) as a reasoning problem, (ii) Japanese DocRE resources and evaluation, and (iii) LLM-based generative information extraction with structured outputs.

DocRE as multi-sentence relation prediction. DocRED established DocRE as a setting where relations often require cross-sentence evidence aggregation and introduced standard evaluation protocols such as micro-averaged F1 Yao et al. (2019). Subsequent syntheses emphasize persistent DocRE challenges—quadratic candidate pairs, coreference/aliasing, and multi-hop reasoning—and the computational cost of global inference Delaunay et al. (2023). What is insufficient for our setting is not the task definition, but the assumption that one can train and deploy specialized DocRE models; we instead target a training-free, low-cost prompting regime where false positives dominate.

Japanese DocRE and JacRED. JacRED fills a resource gap by providing a Japanese Wikipedia DocRE benchmark with evidence sentences and reporting that LLM in-context learning underperforms supervised Japanese models, with particularly low recall on native Japanese text Ma et al. (2024). Prior work primarily frames these results as a capability gap between LLM prompting and supervised learning. Our perspective differs: we treat low-cost LLM prompting as a deployment constraint and ask how to make its outputs more trustworthy, even when raw recall is limited.

Generative information extraction and verification. Generative IE uses LLMs to produce structured predictions directly but is prone to hallucination and format instability Xu et al. (2023). This line identifies constrained decoding and post-hoc filtering as practical mechanisms for reliability, but often discusses them as generic tools. We operationalize these ideas specifically for DocRE by introducing an explicit proposal–verification decomposition designed to suppress over-generation at document scope, rather than relying on a single generation pass.

Schema-constrained structured outputs. Recent work benchmarking JSON-schema constrained decoding shows that constraint engines can yield high compliance and affect downstream quality Geng et al. (2025). This literature largely evaluates constraint mechanisms themselves. Our conceptual difference is pipeline-level: we use structured outputs as an enabling interface that makes verification and deterministic filtering feasible and cheap in practice, focusing on DocRE precision control rather than proposing a new constrained decoding algorithm.

In summary, existing DocRE work explains why document scope is hard, Japanese DocRE work establishes that LLM prompting lags supervised models, and generative IE work motivates constraints for reliability. What remains underexplored is a deployable, low-cost workflow that directly targets the dominant practical failure mode of one-shot LLM DocRE—spurious triples—under a realistic Japanese benchmark.

3 BACKGROUND

DocRE task definition. A document D consists of a sequence of sentences and a set of entities E = {e1, . . . , en}, where each entity aggregates one or more mentions in D. Each entity ei has a type ti from a fixed type set. The goal is to predict a set of directed relation instances R_{hat}, where each instance is a triple (h, r, t) with head entity $h \in E$, relation label r from a predefined relation set, and tail entity $t \in E$.

Dataset: JacRED. JacRED is a Japanese Wikipedia-based DocRE dataset with 2,000 documents and 42,241 labeled relation triples over 35 relation labels, split into train/dev/test = 1400/300/300 Ma et al. (2024). Entities are annotated with 8 types (IREX-style). Each labeled relation is accompanied by evidence sentences, which support analysis of where in the document the relation is grounded.

Evaluation protocol. We evaluate predicted relation triples against gold triples using micro-averaged precision, recall, and F1 aggregated over relation instances across the evaluated documents. Let TP, FP, and FN denote counts of true positives, false positives, and false negatives under exact match of (head, relation, tail). Precision = TP/ (TP+FP), Recall = TP/ (TP+FN), and F1 = 2PR/ (P+R).

4 METHOD

Conceptual overview. Low-cost LLMs prompted for DocRE often fail in a predictable way at document scope: they output many plausible but unsupported triples (over-generation), and unconstrained text generation can introduce format errors that hinder downstream use. We address these failure modes by decomposing extraction into (1) a broad proposal step and (2) a focused verification step, while enforcing structured outputs to make the pipeline deterministic and auditable.

Candidate proposal (what is generated). Given a document D, the proposal stage produces a set of candidate relation triples C. This stage is intentionally recall-oriented: it aims to surface many potentially true relations, accepting that some will be wrong. Outputs are constrained to a JSON Schema so that each candidate has a consistent representation (head, relation, tail) suitable for automated processing.

Candidate verification (how it is validated). The verification stage takes D and the candidate set C and decides, for each candidate triple, whether the relation is supported by the document. This step is precision-oriented and is run in batches to reduce overhead. Verification outputs are also schema-constrained to ensure that each candidate receives an explicit, machine-readable decision.

Type-pair filtering (why an extra guardrail exists). Some predicted triples are structurally implausible because a relation rarely (or never) holds between certain entity-type pairs. To suppress such errors without additional model calls, we apply a deterministic filter based on relation-specific domain and range statistics computed from the training set: a verified triple is removed if its (head_type, tail_type) pair was never observed for that relation in training.

Pipeline definition. Let G (D) denote the proposal stage that returns candidates C. Let V (D, C) denote the verification stage that returns the accepted subset $C_{\text{ver}} \subseteq C$. Let F $(\cdot)$ denote deterministic type-pair filtering. The final prediction is $R_{\text{hat}} = F(V(D, G(D)))$.

Design rationale and terminology consistency. Throughout, we refer to Stage 1 as proposal (candidate generation) and Stage 2 as verification. Proposal targets coverage under schema constraints (mitigating format failures), while verification targets false positive reduction by forcing a support judgment conditioned on the document (mitigating over-generation). The final filter targets predictable type mismatch errors independent of model behavior.

5 EXPERIMENTAL SETUP

Evaluation scope and validity constraints. We evaluate on the JacRED development set Ma et al. (2024) but restrict to 10 documents to control API cost and allow repeated runs across model configurations. Because this subset is small, our claims focus on qualitative robustness (error-type shifts) and consistency across settings rather than precise estimates of dataset-wide performance.

Data selection. The 10 documents are selected by stratified sampling over character length, ensuring that the evaluation includes both shorter and longer documents. This reduces the risk that results reflect a narrow length regime.

Compared conditions (what varies). We compare: - Baseline: one-shot LLM extraction in a single call. - Proposed: proposal–verification pipeline with schema-constrained outputs and deterministic type-pair filtering.

Model configurations. We test Gemini 2.0 Flash, Gemini 2.5 Flash, and Gemini 3 Flash Preview. For Gemini 2.5 Flash and Gemini 3 Flash Preview, we additionally test "thinking" off vs. a thinking budget of 2048; Gemini 2.0 Flash is run without thinking.

What is held constant. For each (model, thinking) configuration, baseline and proposed methods run on the same document subset and are scored with identical matching rules and micro-averaged metrics.

Metrics. We report micro-averaged precision, recall, and F1 over relation triples, along with TP/FP/FN counts, using the standard DocRE formulation Yao et al. (2019).

Limitations introduced by design. The small, non-exhaustive dev subset increases variance and limits generalizability. In addition, the proposed pipeline uses more model calls than the baseline; therefore, comparisons should be interpreted as evidence about reliability/precision control under a low-cost regime, not as a pure measure of single-call model capability.

6 RESULTS

We summarize results by cross-configuration trends, then highlight notable trade-offs.

Observations (consistent patterns). Across all Gemini Flash configurations, the proposed pipeline reduces false positives relative to the one-shot baseline. This yields a consistent precision gain in every setting (baseline precision 0.17–0.31 vs. proposed precision 0.30–0.37). In most configurations, this precision gain translates into higher micro-F1, with the largest improvement observed for Gemini 2.0 Flash (+0.08 F1) and for Gemini 2.5 Flash with thinking 2048 (+0.10 F1).

Observations (recall and F1 variability). Recall remains low in all settings (0.12–0.22), and F1 does not always improve. In particular, for Gemini 2.5 Flash with thinking off, the proposed method sharply reduces FP ($115 \rightarrow 42$) but also reduces TP ($24 \rightarrow 18$), leaving F1 unchanged. A similar effect occurs for Gemini 3 Flash Preview with thinking 2048, where precision improves but recall drops slightly, resulting in no F1 gain.

Interpretations (hypotheses consistent with the data). The uniform FP reduction supports the hypothesis that one-shot DocRE primarily fails by over-generation, which proposal–verification directly targets. The occasional TP loss suggests that the verifier and/or deterministic filtering can become overly conservative, especially under certain model settings, rejecting borderline but correct relations.

Takeaway. The main robustness result is precision control: verification reliably suppresses spurious triples across models. The main remaining bottleneck is recall, indicating that improving candidate coverage and evidence grounding is more important than further filtering if the goal is higher end-to-end F1.

7 CONCLUSION

Low-cost LLM prompting for Japanese DocRE is attractive for deployment but tends to over-generate unsupported relation triples at document scope, limiting precision. We addressed this practical failure mode by decomposing extraction into proposal and verification with schema-constrained structured outputs, supplemented by a lightweight deterministic type-pair filter.

On a stratified subset of JacRED development documents, the approach consistently reduces false positives and improves precision across all tested Gemini Flash configurations, while improving micro-F1 in most settings. At the same time, overall recall remains low and, in some configurations, increased conservatism reduces true positives.

The broader implication is that reliability for knowledge graph construction can be improved without model training by adding an explicit verification layer around LLM extraction. Future work should focus on recall-oriented improvements—better candidate enumeration, stronger evidence localization, and Japanese-specific discourse handling—while preserving the precision benefits of verification.

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.

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.