SCHEMA-CONSTRAINED TWO-STAGE VERIFICATION FOR LOW-COST JAPANESE DOCUMENT-LEVEL RELATION EXTRACTION

Anonymous authors
Paper under double-blind review

ABSTRACT

Document-level relation extraction (DocRE) is a key step for building knowledge graphs from text, but in a low-cost setting a single large language model (LLM) call often over-generates plausible relations that are not supported by the document, yielding low precision, especially for Japanese. This is difficult because DocRE requires cross-sentence reading, documents induce many entity pairs, and free-form generation is brittle to parse and filter. We introduce a Two-Stage pipeline that enforces schema-compliant JSON outputs, separates recall-oriented candidate generation from precision-oriented verification, and applies deterministic relation-specific type-pair constraints derived from JacRED training data. On 10 character-length–stratified JacRED dev documents, across Gemini Flash variants (2.0/2.5/3 preview, with and without "thinking"), logged micro-averaged results show consistent false-positive reductions and precision gains in every configuration, with F1 improving in most and reaching a best observed 0.27. These findings suggest that even without task-specific training, combining structured decoding with explicit verification and lightweight constraints can make practical Japanese DocRE more reliable.

1 INTRODUCTION

Document-level relation extraction (DocRE) aims to identify semantic relations between entities mentioned in a document, where evidence may span multiple sentences and require discourse-level reasoning. DocRE has become a standard formulation for building structured knowledge from text, with widely used benchmarks such as DocRED demonstrating that a substantial fraction of facts require multi-sentence evidence and that naïve sentence-level approaches are insufficient **?**. More recently, Japanese DocRE has gained attention due to the scarcity of native resources and the mismatch between translated corpora and Japanese discourse phenomena. JacRED addresses this gap by providing a Japanese Wikipedia-based benchmark with evidence sentences and a relation schema tailored to Japanese documents **?**.

Despite the progress of supervised DocRE models on gold corpora, practitioners increasingly explore large language models (LLMs) as low-cost extractors to quickly build knowledge graphs without training or with minimal prompting. Surveys of generative information extraction emphasize the appeal of prompting and constrained decoding for structured outputs, but also highlight persistent issues with hallucinations and format faithfulness **?**. In DocRE, these issues are amplified by the quadratic number of entity pairs per document, ambiguous coreference and aliasing, and the need to connect distant mentions; comprehensive reviews of DocRE identify these as central obstacles that affect both discriminative and generative methods **?**.

This paper studies a practical, low-cost setting: Japanese DocRE triple extraction using the Gemini Flash family. Our starting hypothesis is that one-shot extraction (a single LLM call that simultaneously proposes entities and relations) yields low precision due to a large number of false positives. In our preliminary logs, this failure mode is dominant: the baseline tends to over-generate plausible but unsupported relations, and its free-form outputs complicate deterministic filtering.

We propose a simple but principled remedy: a Two-Stage pipeline that (i) forces structure at the interface via JSON Schema–constrained decoding, (ii) decomposes the task into recall-oriented candidate generation and precision-oriented verification, and (iii) applies a deterministic, data-driven constraint layer based on relation-specific allowable (head_type, tail_type) pairs derived from training data. Our design is motivated by two observations. First, structured decoding can substantially increase compliance with a desired schema and can reduce downstream errors by preventing invalid outputs **?**. Second, verification prompts can act as a precision filter when the generator is encouraged to be permissive.

We empirically validate the approach on 10 documents sampled from the JacRED dev set using character-length–based stratified sampling. We compare (a) a Baseline one-shot extractor to (b) the Proposed Two-Stage pipeline across three model families (Gemini 2.0 Flash, 2.5 Flash, and 3 Flash Preview) and multiple thinking configurations (none/off versus a 2048 setting where available). We evaluate micro-averaged precision, recall, and F1 over the 10 documents.

Across all tested configurations, the proposed pipeline consistently reduces false positives (FP), which translates into substantial precision improvements. The best precision gains are observed when verification and type constraints are combined, yielding up to 0.37 precision in our logs. F1 improves in most cases, with best observed F1 of 0.27; in some configurations F1 remains unchanged because recall drops when verification is overly conservative. Overall, the results support the central claim: even with low-cost LLMs, practical accuracy improvements are achievable by decomposing extraction into generation and verification and by applying deterministic constraints.

Contributions: - We formulate a low-cost Japanese DocRE extraction setting on JacRED and empirically characterize the dominant error mode of one-shot LLM extraction as false-positive overgeneration. - We propose a Two-Stage extraction pipeline that uses JSON Schema–constrained decoding for both candidate generation and batch verification, improving reproducibility and enabling deterministic post-processing. - We introduce a lightweight, training-data-derived domain and range constraint filter based on empirically observed (head_type, tail_type) pairs per relation. - We provide an empirical comparison across Gemini Flash family configurations showing consistent FP reductions and precision gains, with best observed F1 of 0.27 on 10 JacRED dev documents.

Future work should primarily target recall. Our analysis suggests that entity alignment robustness, relation directionality errors, and implicit or multi-hop relations remain key bottlenecks; addressing these may require improved prompts, better decomposition strategies, or integrating evidence selection as suggested by prior DocRE research **?**.

2 Related Work

Our work intersects three research threads: document-level relation extraction, Japanese DocRE resources, and LLM-based generative information extraction with structured decoding.

DocRE benchmarks and supervised modeling. DocRED established DocRE as a benchmark where many relations require cross-sentence reasoning and where evaluation emphasizes micro-averaged F1 over multi-label relations per entity pair **?**. The survey by Delaunay et al. synthesizes subsequent progress, comparing sequence, graph, and transformer-based DocRE models and emphasizing core challenges such as quadratic candidate pairs and sensitivity to upstream entity and coreference quality **?**. Compared to these supervised approaches, our setting is deliberately low-cost and training-free: we do not train DocRE models, but instead study prompt-driven extraction and post-processing. Consequently, state-of-the-art supervised systems on JacRED reported by Ma et al. are not directly comparable under our constraints; they assume access to Japanese encoders, full training splits, and multi-seed training protocols **?**. Our goal is instead to improve precision in an LLM-only pipeline.

Japanese DocRE datasets. JacRED is the first public Japanese Wikipedia-based DocRE benchmark with evidence sentences and a moderate-size relation schema, designed to reflect Japanese-native discourse and topic distributions **?**. Our experiments use the JacRED dev set, but only a small, stratified-by-length sample of 10 documents. This differs from JacRED's standard evaluation over hundreds of documents and multiple random seeds, so our reported numbers should be interpreted as a controlled pilot study rather than a leaderboard result.

LLMs for generative information extraction. Xu et al. survey generative IE methods and discuss prompt-based extraction, constrained decoding, and reliability issues such as hallucinations and format deviation **?**. Our baseline corresponds to a straightforward prompt-based, one-shot generative

IE approach. The proposed method aligns with the survey's reliability agenda by introducing explicit verification and hard constraints.

Structured outputs and constrained decoding. Recent work has systematically benchmarked JSON Schema–constrained decoding engines and found that constrained decoding can provide high schema compliance and sometimes improve downstream task accuracy, though coverage and behavior vary across engines **?**. We adopt this principle operationally: we require the LLM to output JSON matching predefined schemas (EXTRACTION_SCHEMA and VERIFICATION_SCHEMA) so that parsing is deterministic and verification can be automated. Our paper differs from prior constrained-decoding studies in that we do not benchmark decoding engines; instead, we use structured outputs as an enabling mechanism for a DocRE pipeline.

Verification and constraint-based filtering. While classic DocRE work often incorporates constraints implicitly through model architectures or global inference **?**, our approach uses an explicit, deterministic filter based on relation-specific type pairs derived from training data. This is closest in spirit to rule-based or logic-guided post-processing, but adapted to a generative pipeline: the LLM proposes and verifies candidates, then a type constraint layer enforces a lightweight ontology. Methods that require joint modeling of NER and RE, such as end-to-end sentence-level joint extraction with biaffine scoring **?**, are not applicable to our document-level Japanese setting because they assume supervised training and are designed for intra-sentence relations, whereas our pipeline targets document-level triples and uses no task-specific training.

In summary, our work contributes a pragmatic bridge between the DocRE literature's emphasis on long-context reasoning and the generative IE literature's emphasis on controllable outputs: we show that a simple decomposition plus structured interfaces and deterministic constraints can substantially reduce false positives in low-cost Japanese DocRE.

3 Background

Document-level relation extraction. We consider DocRE as extracting directed relational triples from a document: (h, r, t), where h is the head entity, t is the tail entity, and r is a relation label from a fixed schema. A document can express multiple relations for the same entity pair and relations may be supported by evidence spanning multiple sentences **?**. The standard difficulty in DocRE is that the number of possible entity pairs grows quadratically with the number of entities, and that correctly identifying relations requires resolving coreference and aggregating information across the document **?**.

Japanese DocRE and JacRED. JacRED provides Japanese Wikipedia documents annotated with entities, relations, and evidence sentences, addressing the gap that translation-based transfer corpora do not adequately capture Japanese-native discourse and topics **?**. Our experiments use the JacRED development set, but only a subset of 10 documents selected by character-length–based stratified sampling. JacRED defines a set of entity types (8 types following an IREX-style taxonomy) and relations (35 relations, including inverse relations added post hoc) **?**. In our pipeline, entity types are used both for output structure and for the domain and range constraints described below.

Generative information extraction and structured outputs. In generative IE, an LLM is prompted to output structured representations of entities and relations, typically as JSON, tables, or other templates **?**. However, unconstrained generation can yield invalid or inconsistent formats, making evaluation and downstream use fragile. Constrained decoding addresses this by restricting the next-token choices to those that keep the output within a grammar or JSON Schema, increasing schema compliance and enabling reliable parsing **?**.

Problem setting and evaluation. Let D be a set of documents and $G(d)$ the set of gold relation instances for document d in JacRED. Given a system that outputs predicted relation instances $P(d)$, we compute micro-averaged counts over a document set S: TP is the number of predicted triples that match gold, FP is predicted triples not in gold, and FN is gold triples not predicted. Precision is TP / (TP + FP), recall is TP / (TP + FN), and F1 is 2PR / (P + R), computed after aggregating TP, FP, and FN across the evaluated documents. Our primary metric is micro-averaged precision/recall/F1 over 10 documents.

Assumptions. We make three operational assumptions specific to our low-cost setting. First, we treat the LLM as a black-box generator and verifier, without gradient-based training. Second, we require the model to produce outputs that validate against fixed JSON schemas (one for extraction and one for verification), enabling deterministic parsing. Third, we leverage training data only to derive relation-specific allowable (head_type, tail_type) pairs, which we apply as deterministic constraints at the end of the pipeline. This constraint layer is data-driven but does not learn parameters beyond enumerating observed type pairs.

4 Method

Overview. We propose a Two-Stage knowledge graph triple extraction pipeline for Japanese DocRE. The pipeline is designed to reduce false positives produced by one-shot LLM extraction by separating generation from verification and by enforcing deterministic structural and type constraints.

Stage 1: recall-oriented candidate generation. Given a document d, the system prompts an LLM to extract candidate entities and relation triples. The extraction is recall-oriented: the prompt encourages proposing plausible relations rather than being conservative. Crucially, the output is produced via JSON Schema–constrained decoding using an EXTRACTION_SCHEMA, ensuring the result is a valid JSON object with fields necessary for downstream verification and scoring. This structured interface is motivated by evidence that schema-constrained generation improves compliance and reliability for structured tasks **?**.

Stage 2: precision-oriented verification in batches. Stage 1 outputs a set of candidate triples $C(d)$. We then verify candidates using a second LLM call that is explicitly framed as a verification task: for each candidate triple, the model judges whether the relation is supported by the document. To reduce cost, candidates are verified in batches with batch size 10. Verification outputs are constrained by a VERIFICATION_SCHEMA to ensure deterministic parsing of the model's decisions. This stage is intended to filter out unsupported hallucinated relations, the dominant failure mode observed in one-shot extraction.

Deterministic post-processing with domain and range constraints. After verification, we apply a final deterministic filter based on relation-specific domain and range constraints. For each relation r, we compute the set of empirically observed (head_type, tail_type) pairs from the JacRED training data. At inference time, a verified triple (h, r, t) is retained only if (type(h), type(t)) is in the allowable set for r. This implements a lightweight ontology check that is data-driven rather than manually curated.

Formalization. For document d, Stage 1 yields candidates $C(d)$. Stage 2 defines a verifier function V that maps each candidate c in $C(d)$ to a binary decision $V(c, d)$ in $\{0, 1\}$. Let $C^+(d) = \{c \in C(d) : V(c, d) = 1\}$. The type-constraint filter defines a predicate $T(c)$ that is 1 if c's (head_type, tail_type) pair is allowable for the relation label and 0 otherwise. The final prediction set is $P(d) = \{c \in C^+(d) : T(c) = 1\}$. This decomposition explicitly targets precision: each step monotonically filters candidates.

Rationale and expected behavior. In a one-shot setting, the generator must simultaneously (i) identify entities, (ii) select relation labels, (iii) ensure directionality, and (iv) maintain output format, all under token and attention constraints. Our design reduces this burden by shifting conservatism to later stages: Stage 1 maximizes coverage under strict formatting; Stage 2 and type constraints remove unsupported or implausible outputs. This aligns with broader IE observations that constrained decoding and multi-step prompting can mitigate format errors and hallucinations **??**.

5 Experimental Setup

Dataset and sampling. We evaluate on the JacRED development set **?**. To keep costs low while spanning a range of document complexities, we select 10 documents via character-length–based stratified sampling. We perform document-level relation extraction and evaluate micro-averaged precision, recall, and F1 aggregated across the 10 documents.

Systems compared. We compare two methods: (1) One-shot Extraction (Baseline): a single LLM call that extracts entities and relations simultaneously, followed by simple filtering. (2) Two-Stage KG Extraction (Proposed): Stage 1 candidate generation followed by Stage 2 verification, then deterministic filtering via relation-specific domain/range constraints (allowable head and tail type pairs) learned from JacRED training data.

Models and configurations. We test across the Gemini Flash family using the Gemini API: - Gemini 2.0 Flash - Gemini 2.5 Flash - Gemini 3 Flash Preview For 2.5 Flash and 3 Flash Preview, we include two configurations: thinking off and thinking set to 2048. For 2.0 Flash, the logged configuration uses thinking none.

Implementation details. The pipeline is implemented using the Google GenAI SDK and relies on Structured Outputs with JSON Schema–constrained decoding. Stage 1 uses an EXTRACTION_SCHEMA; Stage 2 uses a VERIFICATION_SCHEMA. Verification is performed in batches of size 10. After verification, we apply a deterministic filter that removes triples violating empirically observed (head_type, tail_type) constraints per relation. The final output is scored against JacRED gold triples.

Evaluation metrics. We compute micro-averaged precision, recall, and F1, defined as: Precision = TP / (TP+FP), Recall = TP / (TP+FN), and F1 = 2*Precision*Recall / (Precision+Recall). We also report TP, FP, and FN counts from the logged runs for transparency and to analyze error profiles.

Fairness and comparability. All methods are evaluated on the same 10 documents and use the same underlying model configuration within each comparison pair (Baseline vs Proposed). The Proposed method incurs additional LLM calls for verification; therefore, comparisons should be interpreted as accuracy-focused rather than latency- or cost-normalized. Within the Proposed pipeline, the verification policy and the final type-constraint filter are fixed across models to isolate the effect of decomposition and structured interfaces.

6 Results

We report only results present in the experiment logs for micro-averaged DocRE over 10 JacRED dev documents. For each model and thinking configuration, we compare the Baseline one-shot method to the Proposed Two-Stage pipeline. Across all runs, the Proposed method reduces FP substantially, confirming that verification and type constraints act as an effective precision filter. However, recall remains low (0.12–0.22) in both methods, indicating that missing gold relations are the dominant remaining bottleneck.

Experiment 1: Gemini 2.0 Flash (thinking none). Baseline achieves precision 0.20, recall 0.15, and F1 0.17 with TP=22, FP=86, FN=126. Proposed improves precision to 0.35 and recall to 0.19, yielding F1 0.25 with TP=28, FP=51, FN=121. This is a clear improvement in both precision and F1, driven primarily by a 40.7

Experiment 2: Gemini 2.5 Flash (thinking off). Baseline scores precision 0.17, recall 0.16, F1 0.17 (TP=24, FP=115, FN=124). Proposed improves precision to 0.30 but recall drops to 0.12, resulting in F1 0.17 (TP=18, FP=42, FN=130). Here, the FP reduction is large (115 to 42), but a drop in TP causes no net F1 gain.

Experiment 3: Gemini 2.5 Flash (thinking 2048). Baseline scores precision 0.18, recall 0.17, F1 0.17 (TP=25, FP=115, FN=123). Proposed achieves precision 0.36, recall 0.21, F1 0.27 (TP=31, FP=54, FN=117). This is the best observed F1 in the logs (0.27), with both higher TP and much lower FP than the baseline.

Experiment 4: Gemini 3 Flash Preview (thinking off). Baseline reaches precision 0.26, recall 0.16, F1 0.20 (TP=24, FP=70, FN=124). Proposed improves to precision 0.36, recall 0.22, F1 0.27 (TP=32, FP=56, FN=116). This configuration offers a strong cost-performance trade-off in the accompanying analysis report: it matches the best F1 (0.27) without enabling thinking, while maintaining substantial FP reduction.

Experiment 5: Gemini 3 Flash Preview (thinking 2048). Baseline obtains precision 0.31, recall 0.22, F1 0.26 (TP=33, FP=74, FN=115). Proposed yields precision 0.37, recall 0.20, F1 0.26 (TP=30, FP=52, FN=118). Precision improves and FP decreases, but TP decreases as well, leaving F1 unchanged.

Summary across experiments. The Proposed pipeline consistently reduces FP across all five comparisons, with FP reductions of 20.0

Ablation discussion (implicit). Because the logged experiments compare only Baseline vs the full Proposed pipeline, we cannot isolate verification from type constraints quantitatively. Nonetheless, the consistent FP reductions align with the intended role of Stage 2 verification and deterministic type filtering.

Limitations. First, evaluation is over only 10 documents, so results have higher variance than standard JacRED evaluation and we do not report confidence intervals because per-document scores are not available in the logs. Second, recall remains low (0.12–0.22), suggesting that candidate generation and entity alignment are the main bottlenecks. The analysis report notes specific issues: entity alignment robustness, relation directionality errors, and difficulty capturing implicit and multi-hop relations. Third, enabling thinking improves baseline performance in some cases but can over-regularize verification in the Proposed pipeline, reducing true positives.

Figures. No figures were recorded in the experiment logs (figures list is empty), so we do not include any figure panels here.

7 Conclusion

This paper examined a low-cost, prompt-based setting for Japanese document-level relation extraction on JacRED, where one-shot LLM extraction suffers from low precision due to many false positives. We proposed a Two-Stage pipeline that combines JSON Schema–constrained structured outputs, a recall-oriented candidate generation step, a precision-oriented batch verification step, and a final deterministic type-constraint filter based on relation-specific allowable (head_type, tail_type) pairs observed in training data.

Across Gemini Flash family configurations evaluated on 10 stratified-by-length JacRED dev documents, the proposed pipeline consistently reduced false positives and improved precision in every setting. F1 improved in most configurations, with the best observed F1 reaching 0.27 and precision up to 0.37. These results support the claim that even without task-specific training, reliable knowledge graph extraction can be made more practical by decomposing extraction into generation and verification and by enforcing lightweight, data-driven constraints.

The primary remaining limitation is low recall, which caps F1 and sometimes causes verification to remove true relations. Future work should focus on improving candidate coverage and alignment, reducing directionality mistakes, and designing prompts or decomposition strategies that better capture implicit and multi-hop relations in Japanese documents.

This work was generated by AIRAS (**?**).