SCHEMA-CONSTRAINED TWO-STAGE LLM EXTRACTION FOR JAPANESE DOCUMENT-LEVEL RELATION EXTRACTION

Anonymous authors
Paper under double-blind review

ABSTRACT

Japanese document-level relation extraction (DocRE) is a practical bottleneck for building knowledge graphs from long documents (e.g., Wikipedia), especially when using low-cost large language models (LLMs) that tend to over-generate relations and thus suffer from low precision. We address this gap by separating relation discovery from relation confirmation in a two-stage extraction pipeline that enforces structured outputs and applies simple, data-derived plausibility constraints. On a small, cost-constrained evaluation over 10 JacRED development documents and multiple Gemini Flash configurations, the proposed pipeline consistently reduces false positives and yields sizable precision gains (roughly 1.1–1.8× relative, depending on configuration), with F1 improving in several settings and reaching 0.27 at best. These results suggest that lightweight verification and constraint-based filtering can make LLM-based DocRE more usable in practice, while highlighting recall as the main remaining challenge. Our contribution is an empirically validated, deployment-oriented recipe for improving precision in Japanese DocRE without model training.

1 INTRODUCTION

Document-level relation extraction (DocRE) asks systems to infer semantic relations between entities mentioned anywhere in a document, often requiring cross-sentence evidence aggregation, coreference resolution, and careful handling of directionality. This capability is scientifically interesting because it tests models' ability to integrate distributed evidence, and it is practically important because DocRE outputs directly support knowledge graph construction, question answering, and annotation assistance.

Despite progress in supervised DocRE, deploying high-quality systems remains difficult when labeled data are limited, domains shift, or language-specific discourse features complicate extraction. Japanese DocRE amplifies these issues: JacRED, a Japanese Wikipedia-based benchmark, shows that models relying on cross-lingual transfer from English datasets can miss many relations in native Japanese documents, reflecting both topic mismatch and structural differences such as frequent argument omission. In many real deployments, these factors coincide with a second constraint: the need to use low-cost inference rather than expensive training or large models.

LLMs offer an attractive alternative because they can be prompted to emit relation triples without task-specific training. However, in our preliminary and logged experiments, one-shot LLM extraction on Japanese DocRE is dominated by false positives: the model proposes many plausible-sounding but unsupported triples. This failure is not incidental. DocRE naturally induces a quadratic space of entity pairs; document-level evidence can be indirect, tempting models to extrapolate; and free-form generation makes it hard to reliably parse and deterministically filter outputs at scale. As a result, naive "extract everything in one pass" prompting often yields outputs that are too noisy to be useful.

We explore a simple direction tailored to this precision bottleneck: explicitly decomposing extraction into (i) a recall-oriented proposal step and (ii) a precision-oriented confirmation step, while enforcing structure so that post-processing is reliable. Concretely, we introduce a two-stage pipeline in which an LLM first generates candidate relation triples under JSON Schema–constrained decoding, then re-evaluates those candidates in a separate verification call, again under schema constraints. We further apply relation-specific type-pair constraints estimated from JacRED training annotations to remove candidates that violate empirically observed domain/range patterns.

We implement this pipeline using Gemini Flash models and evaluate it on 10 JacRED development documents sampled to span document lengths while keeping API cost manageable. Across Gemini 2.0/2.5/3 Flash configurations (with and without "thinking"), the two-stage pipeline consistently reduces false positives and improves precision relative to a one-shot baseline; F1 improves in several settings, but recall remains low overall.

Contributions: - We identify false-positive dominance as the key practical failure mode of one-shot, low-cost LLM extraction for Japanese DocRE under realistic document-level candidate explosion. - We demonstrate that a staged "propose then verify" extraction strategy with schema-constrained structured outputs and simple data-derived type constraints yields robust precision improvements across multiple low-cost LLM configurations. - We provide a small-scale, cost-conscious evaluation on JacRED that clarifies the precision–recall trade-offs of verification and constraint filtering, isolating recall as the primary remaining bottleneck.

2 Related Work

We position this work at the intersection of (i) document-level relation extraction, (ii) LLM-based generative information extraction, and (iii) reliability via structured outputs and constraints.

Document-level relation extraction (DocRE). DocRE benchmarks such as DocRED highlight that many relations require aggregating evidence across sentences and that the number of candidate entity pairs grows quadratically with document entity count. Subsequent surveys systematize recurring failure modes—coreference, long-range dependencies, multi-hop reasoning, and computational cost—primarily in the context of supervised discriminative models. What remains insufficient for our setting is guidance on how to make inference-only, low-cost extraction precise enough to be usable when exhaustive candidate consideration encourages over-prediction.

Japanese DocRE and JacRED. JacRED provides a Japanese Wikipedia-based DocRE benchmark with entity types and a fixed relation schema, motivated by the limited effectiveness of cross-lingual transfer from English DocRE. Prior JacRED results indicate that in-context LLM approaches can perform poorly, underscoring the need for stronger reliability mechanisms. Our work differs in perspective: rather than proposing new encoders, training recipes, or dataset contributions, we focus on a deployment-oriented extraction pipeline that assumes no model training and evaluates behavior under explicit cost constraints and small-sample validation.

LLMs for generative information extraction. Surveys of LLM-based IE emphasize brittleness under prompt-only extraction, including hallucinated facts and sensitivity to output formatting. Many proposed remedies rely on supervised fine-tuning, extensive multi-prompt workflows, or external tools. These assumptions can be mismatched to low-cost settings. We instead study a minimal decomposition—generation followed by verification—intended to curb hallucinations without requiring additional training.

Structured outputs, constrained decoding, and constraint-based filtering. Recent work on JSON Schema–constrained generation shows that structured decoding can dramatically improve compliance and enable deterministic downstream processing, although behavior depends on model and engine support. While structured outputs alone address parsing reliability, they do not directly solve semantic over-generation. Our approach treats schema constraints as an enabler for a second-stage verifier and for transparent, deterministic filtering via relation-specific type constraints; the conceptual novelty is the combination of staged semantic verification and lightweight, data-derived plausibility constraints aimed specifically at precision in DocRE.

3 Background

This section defines the task, the dataset context used in experiments, and the evaluation protocol.

Task definition (DocRE). Given a document D and a set of entities E = {e1,…,eN} annotated with document-level mentions, DocRE predicts a set of directed relation triples $T \subseteq E \times R \times E$, where R is a fixed relation inventory. A triple (h, r, t) is correct if the relation r holds from head entity h to tail entity t according to the dataset's annotation guidelines, potentially supported by evidence spanning multiple sentences. Multiple relations may hold for the same entity pair.

Dataset context (JacRED). JacRED is a Japanese Wikipedia-based DocRE dataset with a fixed relation schema (35 relations) and entity type annotations. It is designed to reflect Japanese discourse phenomena and to reduce reliance on cross-lingual transfer. In this paper, we use a small subset of the development split for cost-limited experiments, and we use the training split only as a source of empirical statistics (e.g., relation–type co-occurrence patterns).

Evaluation protocol. For each document, we compare a system's predicted triple set against gold triples and aggregate counts across documents (micro-averaging). Let TP be the number of predicted triples that match gold triples, FP the number of predicted triples not in gold, and FN the number of gold triples not predicted. We compute precision P = TP/ (TP+FP), recall R = TP/ (TP+FN), and F1 = 2PR/ (P+R). This protocol captures the trade-off central to DocRE extraction systems: limiting false positives while recovering gold relations.

Assumptions. We assume (i) a fixed relation inventory and entity typing scheme provided by the dataset, (ii) access to gold triples for evaluation, and (iii) access to a training split from which corpus-level statistics can be computed. We do not assume task-specific model training or external retrieval.

4 METHOD

We introduce a precision-oriented DocRE extraction pipeline that decomposes "produce relations" into two semantically distinct decisions: proposing candidates broadly, then confirming support conservatively. The pipeline is designed for low-cost LLM settings where one-shot extraction over-produces relations and where deterministic parsing is required for reliable filtering.

Overview. Given a document D, the system outputs a predicted triple set $\hat{T}$. The method has three steps: (1) generate a recall-oriented candidate set, (2) verify candidates with a separate precision-oriented LLM call, and (3) apply a lightweight deterministic plausibility filter based on relation-specific type compatibility.

Step 1 — Candidate generation (what is generated). The model produces a set of candidate triples C. This step is intentionally permissive: it aims to include most plausible relations supported by the document, accepting that many candidates may be wrong. To prevent evaluation-time parsing errors, we require the model to emit candidates in a fixed JSON structure (e.g., a list of records with head, relation, tail, and associated entity types) under schema-constrained decoding.

Step 2 — Verification (how candidates are validated). Candidate triples are re-submitted to the model for a binary support judgment (accept/reject), using a separate prompt that emphasizes precision: accept only if the document provides sufficient evidence, otherwise reject. For efficiency, verification is done in batches (10 candidates per call). This step targets the dominant one-shot failure mode—hallucinated or weakly implied relations—by forcing an explicit second-pass decision rather than relying on a single generative pass.

Step 3 — Type-pair plausibility filtering (why this step exists). Even after verification, LLMs can confuse argument roles or propose relations between incompatible entity types. We therefore apply a deterministic filter derived from the dataset's training split: for each relation r, we collect the set S_r of observed (head_type, tail_type) pairs in training annotations. A verified triple (h, r, t) is retained only if its (type(h), type(t)) $\in S_r$. This step is transparent, cheap, and directly addresses a common DocRE error where a relation label is plausible but the arguments are not.

Output. The final prediction $\hat{T}$ is the subset of verified candidates that also satisfy the relation-specific type-pair constraints. Terminology is consistent across steps: "candidate" refers to Step 1 outputs, "verified" to Step 2 accepted candidates, and "filtered" to the final set after Step 3.

Design link to DocRE challenges. The decomposition explicitly responds to (i) candidate explosion (permit broad proposal, then prune), (ii) document-level ambiguity (require explicit evidence in verification), and (iii) post-processing fragility (use schema-constrained outputs to enable deterministic evaluation and filtering).

5 Experimental Setup

Goal and scope. We test whether the proposed two-stage pipeline improves precision (and consequently F1) over a one-shot extraction baseline under a cost-constrained, inference-only regime.

Data selection and justification. Experiments use 10 documents sampled from the JacRED development split via character-length–based stratified sampling. The goal is to cover short-to-long documents while keeping API cost manageable. This design supports a small-scale validation of precision/recall trends but limits statistical power and generalization.

Compared conditions (what is controlled). We compare: - One-shot baseline: a single LLM call that extracts relation triples directly from the document. - Two-stage pipeline: candidate generation → batch verification → type-pair filtering.

Variables (what is varied). We vary the LLM configuration across low-cost Gemini Flash models and settings: - Gemini 2.0 Flash (thinking: none) - Gemini 2.5 Flash (thinking: off vs 2048) - Gemini 3 Flash Preview (thinking: off vs 2048)

Constants (what is held constant). Across all runs, we use the same 10 documents, the same evaluation script, the same micro-averaged metrics (P/R/F1), and the same verification batch size (10) for the proposed method.

Evaluation. For each configuration and condition, we compute TP/FP/FN aggregated across the 10 documents and report micro-averaged precision, recall, and F1. We do not report latency or cost numerically; therefore, claims about efficiency are qualitative and limited to the motivation for the small-sample evaluation.

Limitations introduced by the design. Because results are based on 10 documents, observed differences may be sensitive to document choice and do not constitute a benchmark-level comparison to prior JacRED systems. The study is intended to isolate whether staged verification and deterministic constraints robustly reduce false positives under low-cost LLM inference.

6 Results

We summarize results by emphasizing consistent patterns across model configurations, then report configuration-level outcomes. Unless noted, comparisons are between the one-shot baseline and the proposed two-stage pipeline on the same 10 documents.

Overall pattern (observation). Across all five evaluated Gemini Flash configurations, the proposed pipeline reduces false positives and increases precision relative to the one-shot baseline. The absolute precision gain ranges from +0.06 to +0.18, corresponding to roughly 1.1–1.8× relative improvement depending on the baseline level.

Overall pattern (observation). Recall remains low in all settings (0.12–0.22). In two configurations, recall drops under the proposed method, indicating that verification and/or type filtering can remove some true positives.

Configuration-level results (observation). - Gemini 2.0 Flash (no thinking): precision improves 0.20→0.35 and F1 0.17→0.25; FP drops 86→51. - Gemini 2.5 Flash (thinking off): precision improves 0.17→0.30 but recall drops 0.16→0.12; F1 remains 0.17; FP drops 115→42. - Gemini 2.5 Flash (thinking 2048): precision improves 0.18→0.36 and recall 0.17→0.21; F1 improves 0.17→0.27. - Gemini 3 Flash Preview (thinking off): precision improves 0.26→0.36 and recall 0.16→0.22; F1 improves 0.20→0.27. - Gemini 3 Flash Preview (thinking 2048): precision improves 0.31→0.37 while recall drops 0.22→0.20; F1 remains 0.26; FP drops 74→52.

Interpretation (hypotheses, not conclusions). The consistent FP reduction supports the claim that a second-pass verification decision and type compatibility checks curb over-generation. The mixed recall behavior suggests that the verifier can be overly conservative or that type-pair constraints may prune valid but rare type combinations; both effects would disproportionately hurt recall in a small evaluation.

Takeaway. The method is robustly precision-improving across low-cost configurations, but overall performance is bounded by missed relations (FN counts remain high), making recall the primary target for future improvements.

7 Conclusion

Low-cost LLM prompting for Japanese document-level relation extraction can be impractical due to pervasive false positives from one-shot extraction. We showed that decomposing extraction into candidate proposal and explicit verification, while enforcing structured outputs and applying simple relation-specific type compatibility constraints, reliably improves precision across multiple Gemini Flash configurations on JacRED.

What this study demonstrates is a consistent reduction of spurious triples under a small, cost-constrained evaluation regime, and in several settings this translates into higher F1. What remains unresolved is recall: many gold relations are still missed, and in some configurations the verification/constraint steps trade away true positives.

More broadly, the results suggest a practical direction for LLM-based DocRE: treat precision control as a first-class design goal via staged decision-making and transparent constraints, then focus subsequent work on recall-oriented improvements (e.g., better entity alignment and directionality handling, and prompts or decompositions that recover implicit document-level relations) without reintroducing false positives.

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