EFSG: Evidence-First Structured Generation for Multilingual RAG Report Generation

Shaurya Gupta
Thapar Institute of
Engineering and Technology
sgupta8_be24@thapar.edu

Jatin Bedi
Thapar Institute of
Engineering and Technology
jatin.bedi@thapar.edu

Abstract

We describe EFSG (Evidence-First Structured Generation), our submission to Task B of the RAG4Reports@ACL 2026 shared task. Standard retrieval-augmented generation pipelines allow generation models to write from parametric memory and attach citations retroactively: a behaviour we term ***post-rationalization***. EFSG addresses this structurally through a ***phase boundary***: all evidence is retrieved, extracted, and sealed into a fact pool before any generation begins; each sentence then sees only its single committed source passage. Our best run (t5,100k doc corpus) achieved sentence_support of 0.612 and nugget_coverage of 0.126 (F1 = 0.182).

1 Introduction

Retrieval-augmented generation (Lewis et al., 2021) improves factual grounding by supplying a language model with retrieved passages at generation time. In practice, however, large language models draw heavily on parametric memory: a sentence is composed first, and a retrieved document is selected afterward to justify it. Because the generation model has access to the full context window - prior sentences, other documents, and background knowledge, it can write a plausible claim and subsequently identify a passage that approximately supports it. Prompting-based approaches, including Self-RAG (Asai et al., 2023), substantially reduce but not eliminate this tendency, since the fundamental access pattern remains unchanged.

EFSG is built on the hypothesis that post-rationalization is a structural problem requiring a structural solution: the citation must be *committed* before the sentence is written, and the generation model must see *only* the committed passage at the moment of writing. These two constraints together prevent post-rationalization architecturally.

This design is motivated by Bereiter and Scardamalia's distinction between Knowledge-Telling and Knowledge-Transforming in expert writing (Bereiter and Scardamalia, 1987). Knowledge-Telling produces text linearly: each sentence triggers the next without a governing plan. Knowledge-Transforming builds a rhetorical goal structure first, then fills it with evidence. Standard RAG is Knowledge-Telling. EFSG attempts Knowledge-Transforming: explicit epistemic goals are established per section, evidence is retrieved against those goals, sealed, and only then is generation permitted. The epistemic contract in C2 - a verb-led statement of what each section must establish, is the concrete realization of this rhetorical goal structure; it governs both what is retrieved and what may be written.

The paper is structured as follows. Section 2 describes the eight-component pipeline. Section 3 describes the corpus infrastructure. Section 4 presents results and analysis. Section 5 discusses the key design tradeoffs.

2 System Description

Figure 1 shows the full EFSG pipeline. Components C1-C4 constitute *Phase 1: Evidence Construction*. Components C5-C8 constitute *Phase 2: Grounded Generation*. The phase boundary is the explicit seal operation at the end of C4; no new evidence can enter after this point.

2.1 C1+C2: Intent Parser and Section Planner

A single Groq LLaMA-3.3-70B call per topic parses the competition topic JSON into a `ReportIntent` and 4-6 `SectionPlan` objects. Each `SectionPlan` contains (i) a section title, (ii) an *epistemic contract*: a verb-led statement of what the section must establish (e.g., *"Establish the clinical evidence that Tai Chi reduces fall incidence in adults over 65"*) and (iii) 2-3 retrieval queries. Character and sentence budgets are derived from

Proceedings of the First Workshop on Multilingual Report Generation via Retrieval Augmented Generation (RAG4Reports), pages 99–102

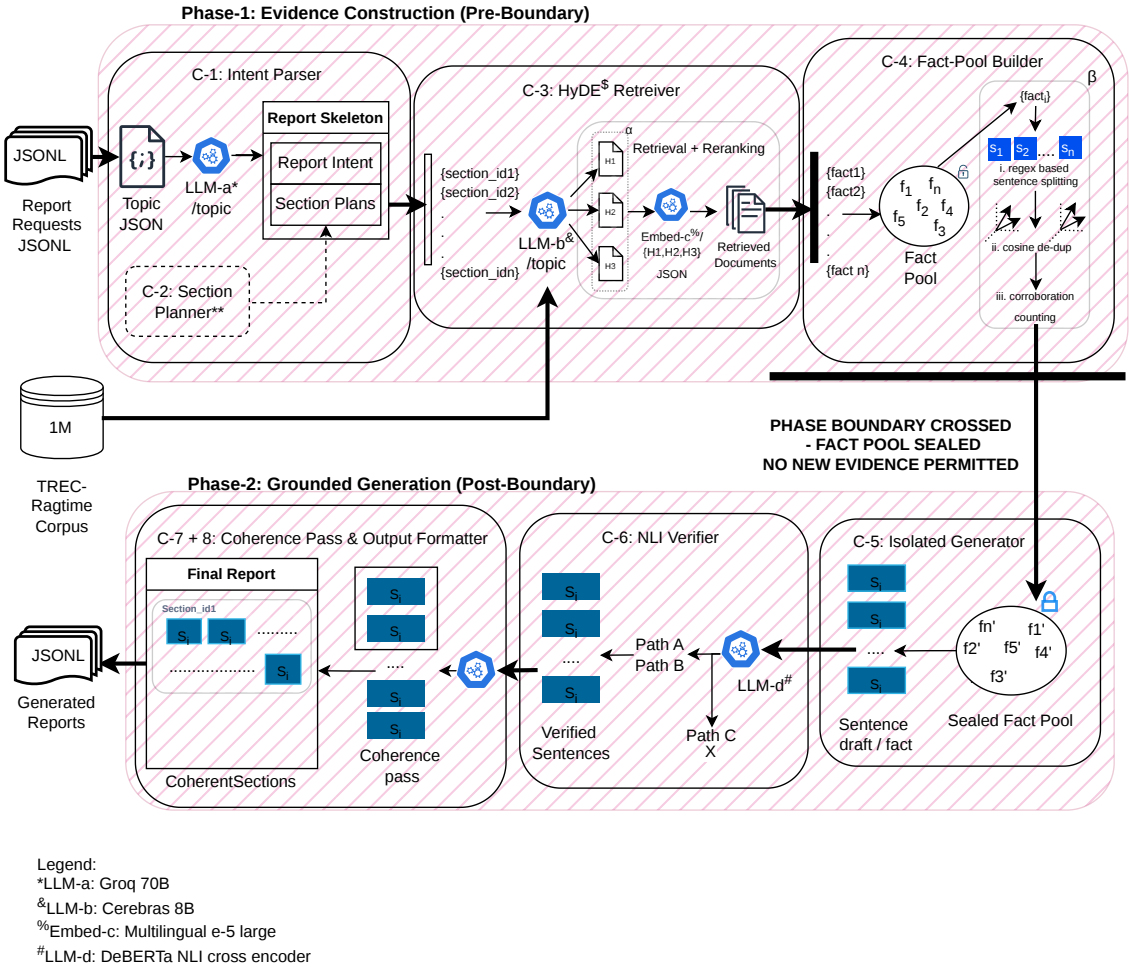

Figure 1: EFSG pipeline. The red line marks the phase boundary. LLM-a: Groq LLaMA-3.3-70B; LLM-b: Cerebras LLaMA-3.1-8B; Embed-c: Multilingual e5-large (text-embedding model); LLM-d: DeBERTa-v3-small NLI cross-encoder (local GPU).

the topic's `limit` field.

These components were originally designed as two separate LLM calls (C1 for intent, C2 for section decomposition). They were merged into one structured JSON response to conserve API quota. **Cost: 1 call/topic.**

2.2 C3: HyDE Retriever

For each section, one Cerebras LLaMA-3.1-8B call generates three hypothetical ideal paragraphs conditioned on the epistemic contract, following the HyDE retrieval strategy (Gao et al., 2022). Each hypothetical is encoded with `multilingual-e5-large` (Wang et al., 2024). The top-20 documents from each hypothetical are pooled (unique by document ID) and re-ranked by cosine similarity against the *epistemic contract embedding* - not the hypothetical. This re-ranking step is deliberate: re-ranking against the contract rather than the hypothetical avoids anchoring retrieval to the model's generated text, instead orienting it toward the section's rhetorical goal. Run logs show that the 8B model occasionally produced malformed JSON, causing fallback to single-hypothetical retrieval for affected sections. **Cost: 1 call/section.**

2.3 C4: Fact Pool Builder (Phase Boundary)

Retrieved documents are segmented into candidate facts using regex sentence splitting. A quality filter discards segments shorter than 30 characters or lacking a common verb form. Within-section semantic deduplication runs at cosine similarity threshold 0.92; cross-section global

deduplication runs at 0.88, preventing the same sentence from appearing in multiple sections. The pool is then **sealed**: `FactPool.sealed = True`. All subsequent components call `pool.get_unused(section_id)`, which enforces the phase boundary by refusing to return facts not present at seal time.

Resource-driven adaptation. The original design called for a Cerebras 8B LLM call per batch of five documents, prompting the model to extract atomic facts conditioned on the section's epistemic contract. At approximately 50 calls per topic for 68 topics, this required roughly 3,400 API calls for C4 alone: infeasible under practical scenarios. Sentence splitting was substituted, reducing C4 to zero API calls. **Cost: 0 API calls.**

2.4 C5: Isolated Generator

Each fact from the sealed pool is used directly as the sentence draft. The committed passage for NLI verification is the fact itself, so NLI(passage, sentence) ≈ 1.0 by construction.

Resource-driven adaptation. The original design called for a per-sentence LLM call that would rephrase the committed passage into appropriate register while remaining informationally contained within it. This step was eliminated to conserve API quota. **Cost: 0 API calls.**

2.5 C6: NLI Verifier

`cross-encoder/nli-deberta-v3-small` (He et al., 2021) runs locally on the T4 GPU. For each sentence draft, the model computes an entailment probability for the (committed passage, sentence) pair. Path A (≥ 0.85): accept as-is. Path B (0.60–0.85): attempt one Groq 70B regeneration with a stricter prompt; keep whichever version scores higher. Path C (< 0.60): substitute the first sentence of the committed passage verbatim. Across the t5 run, 91.9% of sentences took Path A, 8.0% Path B, and 0.1% Path C (1 sentence), with mean NLI = 0.951 and std = 0.042. The near-uniform Path A result is expected given C5's direct-use design: NLI(sentence, sentence) ≈ 1. **Cost: 0 API calls; Path B billed to Groq 70B only when triggered.**

2.6 C7: Coherence Pass

A Cerebras 8B call receives the sentence list for each section and may add minimal transition words (*Additionally*, *However*) at sentence starts only. Rephrasing and factual additions are explicitly prohibited in the system prompt. A post-pass NLI sweep reverts any sentence falling below 0.60. Run logs show that the 8B model frequently produced sentence counts outside the ± 2 tolerance window; for example, splitting one sentence into several or merging adjacent sentences - triggering full section reversion. In the majority of sections across both runs, C7 reverted to the raw C5 output. **Cost: 1 call/section.**

2.7 C8: Output Formatter

Completed sections are serialized to the competition JSONL schema: one object per sentence containing the sentence text, a `citations` field mapping `doc_id` to the NLI score, and metadata (NLI path, fact ID, section title). Character-budget enforcement trims the final section to fit the topic's `limit` field. **Cost: 0 API calls.**

API budget summary. Final configuration: Groq LLaMA-3.3-70B at 4 RPM (C1 only, 1 call/topic); Cerebras LLaMA-3.1-8B at 28 RPM (C3 + C7, ≈13 calls/topic). Total: ≈800 API calls for 68 topics; ≈25 minutes of rate-limit wait. Earlier configurations (with LLM-based C4) required ≈4,100 calls and ≈2.8 hours of wait time across the same 68 topics.

3 Corpus and Infrastructure

The ragtime1 dataset contains 1,000,095 English documents. Documents were sampled directly from the HuggingFace dataset. Documents passing a quality filter (minimum 200 characters, at least three sentences, mean sentence length ≥ 8 words) were embedded using `multilingual-e5-large` and cached to Google Drive. Corpus sizes: 100,000 documents for the t5 run (10% corpus coverage); 50,000 for t4.

4 Results and Analysis

Run	SS	NC	F1
efsg-t5 (100k docs)	0.612	0.126	0.182
efsg-t4 (50k docs)	0.327	0.065	0.097

Table 1: Submission results evaluated with AutoARGUE (Walden et al., 2025). SS = sentence_support; NC = nugget_coverage.

Table 1 shows the two submitted runs. Both metrics scale near-linearly with corpus size: doubling the indexed corpus from 50k to 100k roughly doubles

both sentence_support and nugget_coverage. This scaling pattern isolates the bottleneck as retrieval coverage rather than generation faithfulness.

Sentence_support of 0.612 for t5 is the expected consequence of C5's direct-use design: submitted sentences are verbatim extracts from cited documents, making entailment near-certain by construction. Nugget_coverage of 0.126 reflects document availability: facts absent from the sampled corpus cannot be retrieved regardless of pipeline quality. The two metrics are therefore measuring different failure modes -one of generation, one of retrieval -and should be interpreted independently.

5 Discussion

The central finding is a clean metric decoupling: a system can maximise sentence_support by eliminating generation entirely, while nugget_coverage remains bounded by corpus coverage alone. This suggests that under faithfulness-weighted evaluation, improving retrieval coverage yields larger gains than improving generation quality.

The phase boundary demonstrates that faithfulness guarantees can be structural rather than prompt-based. The invariant: no evidence enters after the pool is sealed holds independently of which models or extraction methods populate the pipeline. Resource constraints affected implementation quality but not the architectural guarantee.

Limitations

Sentence splitting in C4, despite a length and verb-form quality filter remains vulnerable to topical irrelevance, web-boilerplate (cookie notices, navigation text) and non-atomic segments occasionally as direct consequence and trade-off for LLM-based extraction vs API quota.

Direct fact use in C5 means submitted text consists entirely of verbatim source extracts, which maximizes sentence_support by construction but sacrifices fluency and register consistency; C7 was designed to address this through constrained coherence editing, but produced systematic reversion across both runs, leaving most section outputs as raw C5 extracts.

Corpus coverage was limited to 5-10% of the available collection due to compute and storage constraints; this is the primary driver of low nugget_coverage and should be interpreted as a data access limitation rather than a pipeline failure.

References

Akari Asai, Zeqiu Wu, Yizhong Wang, Avirup Sil, and Hannaneh Hajishirzi. 2023. Self-rag: Learning to retrieve, generate, and critique through self-reflection. *Preprint*, arXiv:2310.11511.

Carl Bereiter and Marlene Scardamalia. 1987. *The psychology of written composition*. Lawrence Erlbaum Associates, Inc, Hillsdale, NJ.

Luyu Gao, Xueguang Ma, Jimmy Lin, and Jamie Callan. 2022. Precise zero-shot dense retrieval without relevance labels. *Preprint*, arXiv:2212.10496.

Pengcheng He, Xiaodong Liu, Jianfeng Gao, and Weizhu Chen. 2021. Deberta: Decoding-enhanced bert with disentangled attention. *Preprint*, arXiv:2006.03654.

Patrick Lewis, Ethan Perez, Aleksandra Piktus, Fabio Petroni, Vladimir Karpukhin, Naman Goyal, Heinrich Küttler, Mike Lewis, Wen tau Yih, Tim Rocktäschel, Sebastian Riedel, and Douwe Kiela. 2021. Retrieval-augmented generation for knowledge-intensive nlp tasks. *Preprint*, arXiv:2005.11401.

William Walden, Marc Mason, Orion Weller, Laura Dietz, John Conroy, Neil Molino, Hannah Recknor, Bryan Li, Gabrielle Kaili-May Liu, Yu Hou, Dawn Lawrie, James Mayfield, and Eugene Yang. 2025. Auto-ARGUE: LLM-based report generation evaluation. *Preprint*, arXiv:2509.26184.

Liang Wang, Nan Yang, Xiaolong Huang, Linjun Yang, Rangan Majumder, and Furu Wei. 2024. Multilingual e5 text embeddings: A technical report. *Preprint*, arXiv:2402.05672.