EG-RAG: Retrieval-Augmented Generation with Evidence Graph for Reliable Multi-Document Reasoning

Seunggwan Hong*
KT Corporation
Sungkyunkwan University
South Korea
mygwan112@skku.edu

Junhyung Moon*
KT Corporation
Sungkyunkwan University
South Korea
mjh7345@skku.edu

Eunkyeong Lee
KT Corporation
Sungkyunkwan University
South Korea
ek.lee@skku.edu

Jaehyoung Park
KT Corporation
South Korea
jaehyoung.park@kt.com

Hyunseung Choo†
Sungkyunkwan University
South Korea
choo@skku.edu

ABSTRACT

Retrieval-Augmented Generation (RAG) leverages external knowledge to enhance the recency and factuality of Large Language Models (LLMs), but in real-world retrieval scenarios, ambiguity, contradiction, misinformation, and noise coexist, making RAG vulnerable to arbitrary selection and hallucination propagation. This paper proposes Retrieval-Augmented Generation with Evidence Graph (EG-RAG), which classifies inter-sentence relations as support, contradiction, or neutral using a Natural Language Inference (NLI) classifier to construct an evidence graph. The model then integrates sentence-level relations within relation-specific clusters based on weighted confidence and serializes this structured evidence into the prompt to guide the generation process. EG-RAG first selects a small number of key sentences per document according to query–sentence relevance and defines edge polarity and strength from NLI classification probabilities. Based on these relations, it extracts clusters from relation-specific subgraphs to quantify reliability and directly feeds the structured evidence into the LLM. Across public benchmarks, EG-RAG achieves an average relative performance improvement of 79.09% over standard RAG across diverse backbone models. Notably, it achieves up to 115.63% improvement in conflict reasoning, 95.47% in mixed settings, and 26.17% in multi-hop reasoning. Even when using only a few key sentences per document, the average performance difference remains within 2–3%, confirming high efficiency. In conclusion, EG-RAG achieves reliable reasoning even under ambiguity and noise through hierarchical structuring across sentence, relation, graph, and generation levels, consistently improving both the accuracy and robustness of Retrieval-Augmented Generation.

KEYWORDS

Evidence Graph, RAG, LLM

ACM Reference Format:
Seunggwan Hong, Junhyung Moon, Eunkyeong Lee, Jaehyoung Park, and Hyunseung Choo. 2026. EG-RAG: Retrieval-Augmented Generation with Evidence Graph for Reliable Multi-Document Reasoning. In *Proc. of the 25th International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2026), Paphos, Cyprus, May 25 – 29, 2026*, IFAAMAS, 9 pages. https://doi.org/10.65109/NJIG6104

1 INTRODUCTION

Retrieval-Augmented Generation (RAG) is a representative approach that combines external retrieval to enhance the recency and factual accuracy of Large Language Models (LLMs), whose internal knowledge alone is often insufficient [4, 8, 10]. Early studies that integrated retrieval and generation achieved substantial improvements in knowledge-intensive tasks [5, 9, 16], but real-world retrieval results often contain ambiguous queries, conflicting statements, false or misleading information, and irrelevant noise at the same time. Traditional RAG, which simply merges retrieved documents, is vulnerable to random selection when different documents provide conflicting answers, to frequency bias when identical answers appear multiple times, and to positional bias depending on the location of the correct answer within a document. In particular, the "lost-in-the-middle" phenomenon has been reported, where the model tends to ignore key evidence located in the middle of long inputs [14]. This shows that merely increasing the context length does not ensure reliable reasoning or factual consistency. Instead, long contexts may introduce contradictory or noisy evidence that can mislead the model into generating incorrect answers. Therefore, for RAG to operate reliably, it is necessary to explicitly model the support, contradiction, and neutral relations among retrieved sentences and to integrate them in a structured, relation-centered manner that preserves multiple valid answers.

Existing research addressing this need can be categorized into three directions. First, approaches that control answers through post-generation verification, critique, or self-checking based on retrieval [2, 17, 28]. Second, methods that analyze and mitigate conflicts between internal knowledge and external evidence [23]. Third, frameworks in which document-level agents debate and aggregate to reach consensus [24]. Each of these directions has advantages,

*Equal contribution.
†Corresponding author.
Code Page: https://github.com/AIsysH/eg-rag

Proc. of the 25th International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2026), C. Amato, L. Dennis, V. Mascardi, J. Thangarajah (eds.), May 25 – 29, 2026, Paphos, Cyprus. © 2026 International Foundation for Autonomous Agents and Multiagent Systems (www.ifaamas.org). https://doi.org/10.65109/NJIG6104

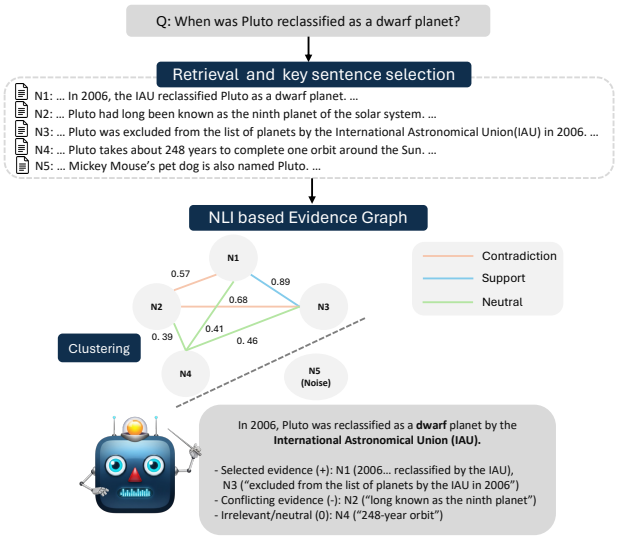

Figure 1: Overview of EG-RAG. Given a question, the retriever gathers relevant documents and selects key sentences. Pairs of selected key sentences are converted into NLI classifier-based edges to form a sentence-level evidence graph. We then perform clustering on this graph, and the resulting cluster information is serialized to guide the LLM in generating the final answer. This example is for key sentence K=1.

but no unified approach has yet been established that structurally represents sentence-level support, contradiction, and neutral relations and connects them consistently to the generation process based on an integrated graph representation. In fact verification, representing inter-sentence interactions as an evidence graph has shown promising results [31], but few studies have extended this idea to the entire retrieval–generation pipeline. Most remain confined to local or post-hoc verification, lacking a unified framework that addresses ambiguity resolution, misinformation suppression, and conflict separation across retrieval, reasoning, and generation.

This paper introduces an evidence graph that classifies inter-sentence relations as support, contradiction, or neutral using a Natural Language Inference (NLI) classifier and assigns confidence-based weights to each relation. The main idea is to apply the principle that a weighted integration of information with different confidence levels leads to rational conclusions even in conflicting environments, across the selection, aggregation, and generation stages of RAG. Specifically, the proposed RAG with Evidence Graph (EG-RAG) in Section 3 operates as follows. First, it selects key sentences based on query-sentence relevance. Second, it defines pairwise relations of selected sentences using NLI classification probabilities and assigns signed edges with magnitudes reflecting confidence. Third, it extracts clusters as connected components from relation-specific subgraphs and constructs reliable evidence sets based on total edge weights. Fourth, it serializes this structure into a prompt that guides the language model to suppress contradictions and reinforce supporting evidence. Details are provided in Section 3.1, and Figure 1 illustrates the process of EG-RAG.

We systematically evaluate EG-RAG on three public benchmarks [18, 24, 29] that cover the challenges of conflict verification, multi-document reasoning, and mixed noise and falsehood. Applying the same pipeline across heterogeneous backbones shows that (i) in conflict-verification and multi-document reasoning tasks, EG-RAG achieves consistent top performance, (ii) under noisy and misinformation-heavy conditions, it maintains robustness, (iii) ablation on the number of key sentences K demonstrates that the gain remains stable even with a small number of sentences ($K = 1 \sim 3$), ensuring efficiency, and (iv) comparison between NLI classifiers and an NLI Agent reveals the generalizability and trade-offs of our design choices. This paper recognizes the limitations of prior RAG research that treated ambiguity, misinformation, and noise separately and presents a new perspective and methodology for addressing them in an integrated manner. The main contributions of this work are as follows.

- **Sentence-level evidence graph** We define a sentence-relation graph with support, contradiction, and neutral edges by combining query-sentence relevance scores and NLI classification probabilities, and we form clusters from relation-specific subgraphs (Section 3.3, 3.4).
- **Weighted integration** We integrate inter-sentence support, contradiction, and neutral relations based on their probabilities and confidence levels across the selection, aggregation, and generation stages. This enables a more effective use of relation-type information, thereby reinforcing multiple evidence sources and ensuring structural transparency in the final response (Section 3.5).
- **Model-invariant and lightweight structuring** The method remains effective with only a few key sentences ($K = 1 \sim 3$), providing high efficiency and reproducible gains across different backbones under the same pipeline (Section 4.2, 4.3).
- **Systematic validation** EG-RAG consistently outperforms strong baselines such as standard RAG, provenance-aware integration, and multi-agent debate, and it quantitatively analyzes the effect of design choices through comparison between NLI classifiers and an NLI Agent (Section 4.4).

Paper Structure. Section 2 reviews procedural approaches and limitations in improving RAG reliability and summarizes related graph representations, benchmarks, and NLI classifiers. Section 3 presents the full EG-RAG procedure: key sentence selection → NLI-based edge weighting → relation-specific clustering → prompt serialization. Section 4 reports experimental settings, comparative results, ablations, and comparisons between NLI classifiers and an NLI Agent. Finally, the conclusion outlines the limitations of this study and potential directions for future research.

2 BACKGROUND

2.1 Procedural Approaches and Limitations for Enhancing RAG Reliability

Post-verification and self-checking To improve the reliability of generated outputs, several approaches have been proposed that self-check the necessity and quality of retrieval or verify the generated results through citation grounding and NLI classifiers. Self-RAG controls retrieval decisions and quality using self-reflection tokens

and performs post-generation verification to reduce unnecessary retrieval and hallucination [2]. Another line of work embeds citation evidence within generated outputs and cross-verifies factuality using an NLI classifier [11]. Although these methods enhance control at the generation stage, they do not structurally represent support, contradiction, and neutral relations among retrieved sentences, thereby leaving cumulative bias unresolved in environments with multiple conflicting signals.

Mitigating conflicts between internal knowledge and external evidence To address inconsistencies between the internal knowledge of LLMs and external retrieval evidence, provenance-aware integration and grouping by consistency or conflict have been proposed. Astute-RAG enhances robustness by analyzing discrepancies between internal and external knowledge through provenance-aware fusion [23]. However, such conflict representations are primarily implemented at the passage level, providing limited mechanisms for quantifying or structurally modeling fine-grained sentence-level conflicts across documents.

Multi-agent debate and aggregation Approaches in which document-level agents collect evidence, engage in debate, and aggregate their conclusions have demonstrated strengths in preserving ambiguity and suppressing falsehoods. For example, MADAM-RAG enables document-level agents to debate and aggregate reasoning, showing effectiveness on datasets with conflicting and ambiguous contexts such as AmbigDocs and FaithEval [24]. Debate-based aggregation is effective in complex and mixed environments but is sensitive to token cost, number of debate rounds, speaking frequency, and strength bias, making consistency difficult to ensure. EG-RAG internalizes the agent functions of evidence decomposition, pairwise relation assessment, and consensus formation within a single-pass graph structure: sentence nodes serve as evidence units, NLI classification acts as pairwise negotiation, and clustering produces consensus without iterative debate, achieving comparable coordination at significantly lower token cost.

Faithfulness-constrained approaches Another direction explicitly constrains factual faithfulness. FaithfulRAG introduces fact-level conflict modeling to detect and adjust discrepancies between retrieved context and the internal knowledge of LLMs [30]. While these approaches aim to maintain contextual faithfulness and prevent false generation, they still do not explicitly graph sentence-level relations. Although each of these methods has strengths, they share a common limitation: none structurally represent support, contradiction, and neutral relations among retrieved sentences at the sentence level or integrate their strengths as weighted signals that connect consistently to the generation process. Chain-of-thought prompting can perform implicit evidence weighing, but still operates on flat passage concatenation and inherits its positional and frequency biases; EG-RAG instead provides pre-structured relations to the LLM. This paper addresses this limitation by introducing an evidence graph that quantifies inter-sentence relations, explicitly modeling fine-grained support and contradiction structures (Section 3).

2.2 Graph Representations for RAG

Evidence graphs in fact verification In fact verification research, including FEVER [22], modeling inter-sentence interactions as an evidence graph has proven effective for multi-evidence reasoning and verification. GEAR represents inter-sentence interactions as a graph, integrating fragmented pieces of evidence to enhance consistency [31]. However, these studies remain limited to the fact verification domain. Few have generalized this concept to the entire retrieval-generation pipeline to handle ambiguity, falsehood, and noise simultaneously while improving performance.

Document and passage-level graph-based RAG In the RAG context, several methods explore document and passage-level graphs to identify evidence paths through structural queries. REALM combines retrieval with pretraining to systematize the use of external knowledge [5]. GRAG searches for optimal subgraphs using a dual view of text and graph representations [7]. HopRAG explores connected chains of evidence through a retrieve-reason-prune loop [13], while SRAG performs multi-entity question answering using structured queries over a knowledge graph [12]. These approaches excel in connectivity and path exploration, but conflicts arising from false or contradictory information still occur at the sentence level, and mechanisms for capturing or reconciling these fine-grained inconsistencies remain limited. Our study adopts an evidence graph where nodes correspond to sentences and edges represent support, contradiction, or neutral relations classified by an NLI classifier. We extract clusters from relation-specific subgraphs and perform weighted integration by combining sentence-level relations to construct reliable evidence sets. This approach preserves and quantifies the connectivity advantages of document- and passage-level graphs, achieving robustness in mixed environments such as FaithEval-Inconsistent [18], HotpotQA [29], and RAMDocs [24]. Unlike MADAM-RAG's explicit multi-round debate, EG-RAG encodes decomposition, negotiation, and aggregation directly into graph edges and clusters, making agent-like coordination implicit yet structurally transparent.

2.3 Benchmarks

We evaluate our method on three benchmarks. Each benchmark captures different challenges commonly encountered in realistic retrieval environments and directly aligns with the goal of achieving reliable RAG reasoning.

FaithEval-Inconsistent This benchmark evaluates whether a model can detect and reconcile conflicts to produce reliable outputs when contradictory contexts are provided [18]. It requires faithfulness and the ability to suppress error propagation under conflicting evidence. This requires separating weakly related information and systematically organize inter-sentence relations.

HotpotQA This benchmark requires multi-document and multi-hop reasoning that connects evidence across documents [29]. The critical factor is maintaining coherence and consistency along reasoning paths. Our approach integrates supporting edges at the sentence-relation level to combine evidence paths that lead to correct answers while excluding weakly related edges.

RAMDocs This benchmark simulates realistic scenarios containing ambiguous queries, false or misleading information, and irrelevant noise, along with an imbalance in the number of supporting documents [24]. When misinformation dominates, cumulative bias grows. Our method models relationships among sentence clusters in the evidence graph to mitigate overall bias.

2.4 NLI Classifier

An NLI classifier categorizes the logical relation between two sentences as support, contradiction, or neutral, serving as the foundation for defining edges and weights in EG-RAG. We leverage classifiers trained on large-scale inference and verification corpora—SNLI [3], MultiNLI [26], ANLI [21, 27], and FEVER [22]—which jointly provide broad coverage of entailment, contradiction, and adversarial reasoning across diverse genres and domains [6, 15].

The resulting support, contradiction, and neutral probabilities are transformed into edges (Equations (6)–(9), Section 3.3) and subsequently used for clustering (Section 3.4) and prompt generation (Section 3.5).

3 RETRIEVAL-AUGMENTED GENERATION WITH EVIDENCE GRAPH (EG-RAG)

3.1 Overview

EG-RAG constructs a sentence-level evidence graph by combining query-sentence relevance and NLI classification, summarizes it into relation-specific clusters, and serializes this structure into a prompt to generate the final answer. The overall procedure is summarized in Figure 1. First, the retriever gathers relevant documents and divides them into sentences. It then selects the global top-K sentences by cosine similarity to the query and adds at least one representative sentence per document to form the key set (Section 3.2). Next, the NLI classification between selected sentences produces support, contradiction, and neutral probabilities, which are used to define signed and weighted edges based on their labels and confidence (Section 3.3). Then, connected components are extracted from relation-specific subgraphs to form clusters, and total edge weights are used to quantify their reliability (Section 3.4). Finally, the query, document-level representative sentences, and relation-specific clusters are serialized into a structured prompt that guides the language model to suppress contradictions and emphasize supporting evidence during answer generation (Section 3.5). This process enables EG-RAG to structurally represent inter-sentence relations and to form the evidence graph.

The example in Figure 1 illustrates the process for the question "When was Pluto reclassified as a dwarf planet?". Five retrieved documents are used to extract representative sentences $N1$-$N5$ to build a graph. Sentences $N1$ and $N3$ directly reference the International Astronomical Union (IAU)'s 2006 decision and thus support the correct answer. Sentence $N2$ provides outdated information describing Pluto as the ninth planet, forming a contradiction with $N1$ and $N3$. Sentence $N4$ is neutral, mentioning an unrelated fact about Pluto's orbital period, while $N5$ introduces irrelevant noise referring to the cartoon character. When the NLI classifier assigns relation labels and confidence scores to edges, $N1$ and $N3$ form a supporting cluster reinforcing the answer, and the cluster containing $N2$ provides contradictory evidence that corrects the generation process. This structured approach improves robustness to ambiguity, falsehood, and noise compared to simple contextual concatenation.

3.2 Key Sentence Selection

Given a query q and a set of M retrieved documents $\mathcal{D} = \{D_m\}_{m=1}^{M}$, each document D_m is divided into sentences, represented as $D_m = \{s_{m,i}\}$. All sentences are combined into a single global index set $\mathcal{S} = \{s_i\}_{i=1}^{N}$, and the document to which each index i belongs is denoted as $\mathrm{doc}(i) \in \{1, \ldots, M\}$. The subset of sentence indices belonging to document m is defined as

$$\mathcal{S}(m) = \{\, i \in \{1, \ldots, N\} \mid \mathrm{doc}(i) = m \,\}. \tag{1}$$

The main notations are as follows: K is the number of global top-K sentences, $\mathcal{I}_K$ is the index set of top-K sentences, $\mathcal{S}_K$ is the set of top-K sentences, $\kappa(m)$ denotes the most relevant sentence index within document m, $\mathcal{K}$ is the set of per-document representative sentences, and N is the total number of sentences across all documents.

Each sentence s_i and the query q are embedded using the function $E(\cdot)$ [25], and ℓ_2 normalization is applied for cosine similarity. The normalized inner product directly yields the corresponding query relevance score r_i for each sentence:

$$\mathbf{u}_i = \frac{E(s_i)}{\|E(s_i)\|_2}, \quad \mathbf{v} = \frac{E(q)}{\|E(q)\|_2}, \quad r_i = \mathbf{u}_i^\top \mathbf{v} \in [-1, 1]. \tag{2}$$

The top-K sentences by r_i are then selected as

$$\mathcal{I}_K = \mathrm{TopK}(\{r_i\}_{i=1}^{N}, K), \qquad \mathcal{S}_K = \{\, s_i : i \in \mathcal{I}_K \,\}. \tag{3}$$

If a tie occurs among equal scores, the indices are sorted in ascending order to maintain reproducibility. However, since the global top-K selection may omit certain documents, at least one representative sentence is additionally included from each document to preserve ambiguity and ensure complete document coverage:

$$\kappa(m) = \arg\max_{i \in \mathcal{S}(m)} r_i \quad \Rightarrow \quad \mathcal{K} = \{\, (m,\, r_{\kappa(m)},\, s_{\kappa(m)}) \,\}_{m=1}^{M}. \tag{4}$$

The final set of selected sentences combines the global top-K and the per-document representatives:

$$\mathcal{I}^* = \mathcal{I}_K \cup \{\kappa(m)\}_{m=1}^{M}, \quad \mathcal{S}^* = \{\, s_i \mid i \in \mathcal{I}^* \,\}. \tag{5}$$

Hence, $\mathcal{S}^*$ contains both globally important sentences ($\mathcal{I}_K$) and per-document key sentences ($\kappa(m)$), and $\mathcal{S}^*$ is the corresponding set of actual sentences. This ensures strong global evidence while preserving document-level representation and diversity. In practice, the steps are: (i) divide each document into sentences, (ii) compute normalized embeddings, (iii) select global top-K sentences, and (iv) merge them with per-document representatives to form $\mathcal{S}^*$, which serves as input for subsequent graph construction and reasoning.

3.3 NLI Classifier and Edge Weighting

We adopt NLI classification as the relation-induction module because it provides calibrated three-way probabilities (support, contradiction, neutral) in a single forward pass, enabling both edge labeling and confidence weighting without additional training. However, the evidence-graph framework is agnostic to the specific relation module: embedding-based similarity thresholding, LLM-as-judge scoring, or learned pairwise classifiers can replace NLI with minimal pipeline change. From the final sentence set $\mathcal{S}^*$, we identify inferential relations between sentence pairs and define them as graph edges. The goal is to explicitly classify whether two sentences support, contradict, or are neutral to each other so that their relational confidence can be reflected in the graph structure. For two sentences $s_i, s_j \in \mathcal{S}^*$, the NLI classifier outputs a logit vector

$\mathbf{z}_{ij} \in \mathbb{R}^3$ (Section 2.4). After applying the softmax function, we obtain:

$$\begin{aligned} \mathbf{p}_{ij} &= \mathrm{softmax}(\mathbf{z}_{ij}) \\ &= \big(p_{ij}^{\mathrm{con}},\ p_{ij}^{\mathrm{neu}},\ p_{ij}^{\mathrm{sup}}\big), \\ &\sum_{\ell \in \{\mathrm{con,neu,sup}\}} p_{ij}^{\ell} = 1. \end{aligned} \tag{6}$$

Here, p_{ij}^{con}, p_{ij}^{neu}, and p_{ij}^{sup} represent the probabilities of contradiction, neutral, and support respectively.

The label and its associated confidence are defined as:

$$\hat{\ell}_{ij} = \arg \max_{\ell \in \{\mathrm{con,neu,sup}\}} p_{ij}^{\ell}, \tag{7}$$

$$\hat{p}_{ij} = \max\{p_{ij}^{\mathrm{con}}, p_{ij}^{\mathrm{neu}}, p_{ij}^{\mathrm{sup}}\}. \tag{8}$$

The final edge weight combines query relevance and confidence:

$$\begin{aligned} w_{ij} &= \hat{p}_{ij}\, r_i\, r_j, \\ \mathrm{label}(i,j) &= \hat{\ell}_{ij} \in \{\mathrm{con,neu,sup}\}. \end{aligned} \tag{9}$$

A larger w_{ij} indicates that both sentences are highly relevant to the query and that the NLI classifier is confident in its relation. Thus, all pairs in $\mathcal{S}^*$ obtain signed edges where $w_{ij} = \hat{p}_{ij}\, r_i r_j$ represents both sign and magnitude.

3.4 Evidence Graph and Clusters

Using these edges, we construct the sentence-level evidence graph:

$$G = (V, E), \qquad V = \mathcal{I}^*.$$

Each node $i \in V$ represents a sentence s_i with its relevance score r_i. Edges follow the labels and weights from Eq. (9). Relation-specific subgraphs are defined as:

$$\begin{aligned} G^{\ell} &= (V, E^{\ell}), \\ E^{\ell} &= \{(i,j) \in V \times V : \hat{\ell}_{ij} = \ell\}, \quad \ell \in \{\mathrm{con,neu,sup}\}. \end{aligned} \tag{10}$$

Thus, G^{con}, G^{neu}, and G^{sup} correspond to contradiction, neutral, and support relations respectively. Each subgraph is decomposed into connected components:

$$\mathcal{C}^{\ell} = \{\, C \in \mathrm{Comp}(G^{\ell}) \mid |C| \geq 2 \,\}, \quad \ell \in \{\mathrm{con,neu,sup}\}. \tag{11}$$

A cluster is retained only if it contains at least two nodes. To quantify cluster strength, we sum all internal edge weights:

$$\Phi^{\ell}(C) = \sum_{(i,j) \in E^{\ell}(C)} w_{ij} = \sum_{(i,j) \in E^{\ell}(C)} \hat{p}_{ij}\, r_i r_j. \tag{12}$$

Here, $E^{\ell}(C)$ denotes the set of edges within cluster C. A larger $\Phi^{\ell}(C)$ indicates higher reliability and stronger connection to the query. The three relation-specific cluster sets $\mathcal{C}^{\mathrm{con}}$, $\mathcal{C}^{\mathrm{neu}}$, and $\mathcal{C}^{\mathrm{sup}}$ serve as structured inputs for the final answer generation. Clusters with high $\Phi^{\mathrm{sup}}(C)$ reinforce correct reasoning paths, while those with high $\Phi^{\mathrm{con}}(C)$ highlight contextual conflicts that reduce reasoning errors. Neutral clusters $\Phi^{\mathrm{neu}}(C)$ help stabilize judgment by providing contextually balanced evidence.

Algorithm 1 EG-RAG inference

Require: Query q; documents $\mathcal{D} = \{D_m\}_{m=1}^{M}$; top-$K$
Ensure: Final answer $\widehat{\mathcal{A}}$ with evidence clusters

Split each D_m into sentence indices $\mathcal{S}(m)$; build global $\mathcal{S} = \{s_i\}_{i=1}^{N}$
Compute normalized embeddings and relevance by Eqs. (2) to obtain r_i
Select global top-K: $\mathcal{I}_K = \mathrm{TopK}(\{r_i\}, K)$; set $\mathcal{S}_K = \{s_i : i \in \mathcal{I}_K\}$
for $m = 1$ to M **do** ▷ per-document representative
  $\kappa(m) \leftarrow \arg\max_{i \in \mathcal{S}(m)} r_i$
end for
$\mathcal{K} \leftarrow \{(m, r_{\kappa(m)}, s_{\kappa(m)})\}_{m=1}^{M}$ (Eq. (4))
$\mathcal{I}^* \leftarrow \mathcal{I}_K \cup \{\kappa(m)\}_{m=1}^{M}$, $\mathcal{S}^* \leftarrow \{s_i : i \in \mathcal{I}^*\}$
Set $V \leftarrow \mathcal{I}^*$
Edges and weights (Eqs. (6)–(9))
for all (i, j) with $i < j$, $i, j \in V$ **do**
  Compute $\mathbf{p}_{ij}$ by Eq. (6)
  $\hat{\ell}_{ij} \leftarrow$ Eq. (7), $\hat{p}_{ij} \leftarrow$ Eq. (8)
  $w_{ij} \leftarrow \hat{p}_{ij}\, r_i r_j$; add labeled edge $(i, j, \hat{\ell}_{ij}, w_{ij})$ (Eq. (9))
end for
Relation-specific graphs and clusters (Eqs. (10)–(12))
for $\ell \in \{\mathrm{con, neu, sup}\}$ **do**
  Build $G^{\ell} = (V, E^{\ell})$ with edges labeled ℓ (Eq. (10))
  $\mathcal{C}^{\ell} \leftarrow \{C \in \mathrm{Comp}(G^{\ell}) : |C| \geq 2\}$ (Eq. (11))
  for all $C \in \mathcal{C}^{\ell}$ **do**
    $\Phi^{\ell}(C) \leftarrow \sum_{(i,j) \in E^{\ell}(C)} w_{ij}$ (Eq. (12))
  end for
end for
Prompt and decoding (Eqs. (13)–(14))
Serialize $x = \Pi\big(q, \mathcal{D}, \mathcal{K}, \mathcal{C}^{\mathrm{con}}, \mathcal{C}^{\mathrm{neu}}, \mathcal{C}^{\mathrm{sup}}\big)$
$\widehat{\mathcal{A}} \leftarrow \arg\max_{y \in \mathcal{G}} f_\theta(y \mid x)$
return $\widehat{\mathcal{A}}$ with $\mathcal{C}^{\mathrm{con}}, \mathcal{C}^{\mathrm{neu}}, \mathcal{C}^{\mathrm{sup}}$

3.5 LLM-guided Answer Finalization

The query, selected sentences, and relation-specific clusters are serialized into a structured prompt that guides the LLM in generating the final answer. The prompt construction function Π takes the following inputs:

- Query q
- Document set $\mathcal{D}$ and per-document representative sentences
- $\mathcal{K}$ (Eq. (4))
- Relation-specific clusters $\mathcal{C}^{\mathrm{con}}, \mathcal{C}^{\mathrm{neu}}, \mathcal{C}^{\mathrm{sup}}$ (Eq. (11))

Formally,

$$x = \Pi\big(q,\ \mathcal{D},\ \mathcal{K},\ \mathcal{C}^{\mathrm{con}},\ \mathcal{C}^{\mathrm{neu}},\ \mathcal{C}^{\mathrm{sup}}\big). \tag{13}$$

The prompt explicitly includes output constraints such as "output only the answer", "no explanations", and "use the specified list format". The language model f_θ defines the conditional distribution over responses given x, and the optimal answer $\widehat{\mathcal{A}}$ within the predefined format set $\mathcal{G}$ is obtained as

$$\widehat{\mathcal{A}} = \arg\max_{y \in \mathcal{G}} f_\theta(y \mid x). \tag{14}$$

Here, $\mathcal{G}$ specifies structured formats, for instance "output answers as a JSON array only". The final output $\widehat{\mathcal{A}}$ may include multiple

answers if ambiguity exists, directly reflecting multiple valid candidates. This stage organizes information from the evidence graph into the prompt, divides it into three relation-specific cluster sets, and enables the model to generate structurally grounded answers that suppress contradictions and emphasize support. Thus, the generation step of EG-RAG performs prompt-level integration that consolidates multiple reliable evidence sources. The complete inference pipeline integrating sentence selection, relation estimation, clustering, and generation control is summarized in Algorithm 1.

4 EXPERIMENTS AND RESULTS

This section comprehensively evaluates the effectiveness of EG-RAG. Section 4.1 describes the experimental setup. Section 4.2 presents the main comparative results. Section 4.3 analyzes the effect of the number of key sentences K. Section 4.4 compares NLI classifiers and Agent-based relation classification.

4.1 Experimental Setup

Benchmarks and task definitions We evaluate our model on the three benchmarks introduced in Section 2.3. FaithEval-Inconsistent measures faithfulness under conflicting or manipulated contexts [18]. HotpotQA requires multi-document and multi-hop reasoning [29]. RAMDocs evaluates realistic scenarios involving ambiguity, false information, and noise [24]. The example types and evaluation purposes of each dataset are summarized in Section 2.3, and this section focuses on quantitative results.

Backbone models and inference environment We use three language model backbones: GPT-4o-mini, Anthropic-Sonnet-4, and Ministral-8B-Instruct. These span three complementary axes: GPT-4o-mini as a cost-efficient commercial API baseline, Anthropic Sonnet-4 as a long-context safety-oriented model for multi-document reasoning stability, and Ministral-8B-Instruct as a single-GPU open-source model to verify model-agnostic applicability. GPT-4o-mini and Anthropic-Sonnet-4 are accessed through official APIs [1, 20], while Ministral-8B-Instruct is executed on a single NVIDIA A100 GPU [19]. All experiments are conducted in a zero-shot setting with temperature fixed to 0 and maximum output length of 2,048 tokens. The number of key sentences K in EG-RAG is set between 1 and 3 depending on computational resources (Section 3.2). Relations between selected sentences are classified using the NLI classifier (Section 2.4, Section 3.3).

NLI classifier and Agent configuration The NLI-based edges follow the resources and procedures described in Section 2.4 and Section 3.3. We use two classifiers: (1) NLI1: RoBERTa-large-MNLI [15], and (2) NLI2: DeBERTa-v3-large trained further on MNLI, FEVER, and ANLI [6]. For comparison, we also include an Agent-based relation classification setup, where GPT-4o-mini performs ternary classification via a procedural prompt.

Compared methods We compare EG-RAG against representative procedural and graph-based approaches (Section 2.1, Section 2.2). The compared methods are as follows:

- No-RAG: Uses only the internal knowledge of the LLM without external retrieval.
- Standard RAG: Concatenates retrieved documents as input without structural modeling.
- EG-RAG w/o Key Sentences: Removes key sentences to isolate the contribution of the evidence graph.
- Astute-RAG [23]: Mitigates conflicts between internal and external knowledge via provenance-aware fusion.
- Faithful-RAG [30]: Models fact-level conflicts to constrain contextual faithfulness.
- MADAM-RAG [24]: Employs document-level agent debate and aggregation to achieve consensus.

The contribution of each EG-RAG component is verified via ablation studies (e.g., w/o sentences). The main evaluation metric is Exact Match (EM), which is 1 only when the predicted answer set exactly matches the ground truth, and 0 otherwise. For evaluation, HotpotQA uses 1,500 sampled examples with seed 42, while FaithEval-Inconsistent and RAMDocs use their official test splits.

4.2 Main Results: Method Comparison

Table 1 shows the results across three benchmarks (FaithEval-Inconsistent, HotpotQA, RAMDocs) and three LLM backbones (GPT-4o-mini, Anthropic-Sonnet-4, Ministral-8B-Instruct). Overall, No-RAG yields the lowest accuracy in all benchmarks, especially on GPT-4o-mini, confirming that internal knowledge alone is insufficient for reliability. The standard RAG baseline improves results across all benchmarks but remains vulnerable to contextual inconsistency and misinformation, performing poorly on FaithEval and RAMDocs. EG-RAG achieves the best performance across all models on FaithEval. Specifically, GPT-4o-mini achieves 55.53%, Anthropic-Sonnet-4 67.20%, and Ministral-8B-Instruct 42.20%, corresponding to relative gains of 115.82%, 110.46%, and 120.60% over the standard RAG, with an average improvement of 115.63%. The ablation variant w/o sentences shows degraded performance compared to EG-RAG, confirming that sentence selection and the evidence graph play crucial roles in maintaining contextual faithfulness. Compared to MADAM-RAG, EG-RAG improves by 26.44%, 23.37%, and 17.88% on the three models, with an average improvement of 22.53%.

On HotpotQA, EG-RAG again achieves the highest performance across all backbones. GPT-4o-mini scores 72.27%, Anthropic-Sonnet-4 78.40%, and Ministral-8B-Instruct 34.37%, outperforming standard RAG by 15.82%, 7.88%, and 54.82%, respectively, with an average improvement of 26.17%. The gap between EG-RAG and its ablation variant excluding key sentences is within 3%, indicating that graph structuring contributes more to multi-document and multi-hop reasoning performance than key sentence selection alone. Compared with the next-best MADAM-RAG, EG-RAG achieves improvements of 9.95%, 6.58%, and 21.75%, averaging 12.76%. This indicates that while multi-agent debate is effective, direct graph-based structuring of sentence-level evidence yields a more efficient reasoning process.

In the noise-heavy RAMDocs benchmark, MADAM-RAG achieves the highest score, as it is specifically designed for this setting by assigning independent agents to each document for multi-round debates and aggregating their outputs. This approach excels at maintaining ambiguity and suppressing misinformation and noise. EG-RAG ranks second, achieving 28.20%, 31.12%, and 24.00% across the three models. Although lower than MADAM-RAG, it consistently outperforms Astute-RAG and Faithful-RAG and achieves an average improvement of 95.47% over the standard RAG. This demonstrates that EG-RAG maintains competitive performance in

Table 1: Main comparison across backbones on FaithEval (Inconsistent, EM↑), HotpotQA (EM↑), and RAMDocs (EM↑). Bold indicates the best number and underline indicate the second-best in each column.

	FaithEval-Inconsistent			HotpotQA			RAMDocs		
Method	GPT-4o-mini	Anthropic Sonnet-4	Ministral 8B-Inst.	GPT-4o-mini	Anthropic Sonnet-4	Ministral 8B-Inst.	GPT-4o-mini	Anthropic Sonnet-4	Ministral 8B-Inst.
No-RAG	5.82	7.87	6.20	17.67	50.27	9.20	3.00	2.30	0.20
RAG	25.73	31.93	19.13	62.40	72.67	22.20	22.80	12.00	11.80
EG-RAG (ours)	**55.53**	**67.20**	**42.20**	**72.27**	**78.40**	**34.37**	28.20	31.12	24.00
EG-RAG w/o Key Sentences	49.73	54.47	35.80	69.20	77.93	31.33	25.02	23.89	18.83
MADAM-RAG	39.33	48.53	26.20	65.73	73.56	28.23	**32.40**	**32.60**	**25.39**
Astute-RAG	31.93	46.93	21.73	58.13	63.73	23.92	14.00	29.49	12.57
FaithfulRAG	24.20	38.96	18.13	53.27	65.07	23.00	13.67	25.60	13.42

mixed environments involving contextual inconsistency, multi-step reasoning, and noisy information, though it may underperform specialized frameworks optimized for specific benchmarks.

Across all datasets, EG-RAG ranks highest on FaithEval and HotpotQA and performs robustly on RAMDocs, demonstrating that sentence-level NLI classification combined with an evidence graph is effective for both multi-step reasoning and resisting noise. For instance, when two documents state conflicting years for the same event, standard RAG arbitrarily selects one, whereas EG-RAG assigns a contradiction edge, separates them into distinct clusters, and lets the LLM output both valid answers.

4.3 Effect on Number of Key Sentences (K)

Figure 2 shows how the number of key sentences K affects performance. When K is too large, redundant sentences increase graph complexity and introduce noise and contradictions. Conversely, small K values enable focused selection, resulting in simpler graphs and more stable NLI-based relation extraction and clustering. EG-RAG therefore uses $K = 1{\sim}3$. Even at $K = 1$, performance improves over standard RAG. For FaithEval, increasing K from 1 to 3 yields 2.97%, 5.33%, and 7.46% gains for GPT-4o-mini, Anthropic-Sonnet-4, and Ministral-8B-Instruct, respectively. Similarly, on HotpotQA, $K = 3$ yields additional improvements of 2.85%, 1.03%, and 14.11%. Overall, performance increases slightly as K grows, showing that EG-RAG benefits from a small number of key sentences while maintaining efficiency. Even with few sentences, supporting edges strengthen correct reasoning, contradictory edges block incorrect paths early, and neutral edges mitigate irrelevant influence. As a result, the LLM maintains path consistency and effectively mitigates the "lost-in-the-middle" positional bias commonly observed in long-context reasoning [14].

4.4 Comparison with NLI Classifiers and Agent Model

Finally, we compare NLI classifiers with the NLI Agent model (Table 2). NLI1 (RoBERTa-large-MNLI) serves as the baseline, while NLI2 (DeBERTa-v3-large-MNLI-FEVER-ANLI) is trained with several additional datasets to further enhance robustness in factual verification and adversarial conditions.

On the FaithEval benchmark, NLI2 consistently achieves the highest performance across GPT-4o-mini, Anthropic-Sonnet-4, and Ministral-8B-Instruct, indicating its superior suitability for contextual faithfulness evaluation. NLI2, trained on MNLI, FEVER, and ANLI, robustly distinguishes support, contradiction, and neutral relations even under noisy or adversarial settings. This confirms that NLI classifier-based modeling is effective for detecting contradictions and suppressing false evidence. In contrast, on HotpotQA and RAMDocs, NLI2 slightly underperforms NLI1, suggesting that while NLI2 excels in faithfulness tasks, the more conservative NLI1 remains stable in complex multi-document or noisy environments.

The Agent-based NLI classification performs lower than both classifiers, indicating that simple prompt-based relation classification is insufficient for precise sentence-level reasoning. However, performance could be improved through more refined prompt engineering, specialized training for conflict or noise detection, or by incorporating criteria such as multi-evidence comparison and contradiction explanation. Agent models fine-tuned with dedicated datasets may further enhance classification accuracy.

Overall, NLI classifier-based modeling proves most effective for tasks involving inconsistency or misinformation such as FaithEval. RoBERTa-large-MNLI provides a strong and stable baseline under mixed and noisy conditions, whereas DeBERTa-v3-large-MNLI-FEVER-ANLI excels at suppressing explicit contradictions and false evidence but may be less robust in complex retrieval scenarios. Future research should explore hybrid approaches that combine the strengths of both and systematically analyze complementary interactions between Agent-based and classifier-based methods.

EG-RAG strengthens reasoning by (1) increasing evidence density through key sentence selection (Section 3.2), (2) maintaining path consistency via graph-based relation modeling (Section 3.3, Section 3.4), and (3) filtering incorrect reasoning through contradiction and neutral signals (Section 3.5). While related to Astute-RAG in addressing internal and external knowledge conflicts, EG-RAG differs from prior work by explicitly constructing edges through NLI classifiers and directly injecting the structured evidence graph into the model input.

5 CONCLUSION

This paper proposed EG-RAG, a framework that addresses ambiguity, contextual conflict, misinformation, and noise in real-world retrieval environments by selecting sentence-level key evidence and organizing NLI-based relations into an evidence graph. The core idea is to integrate inter-sentence support, contradiction, and neutral relations as weighted signals of consistency, reliability, and multi-answer preservation, mapping them across the selection,

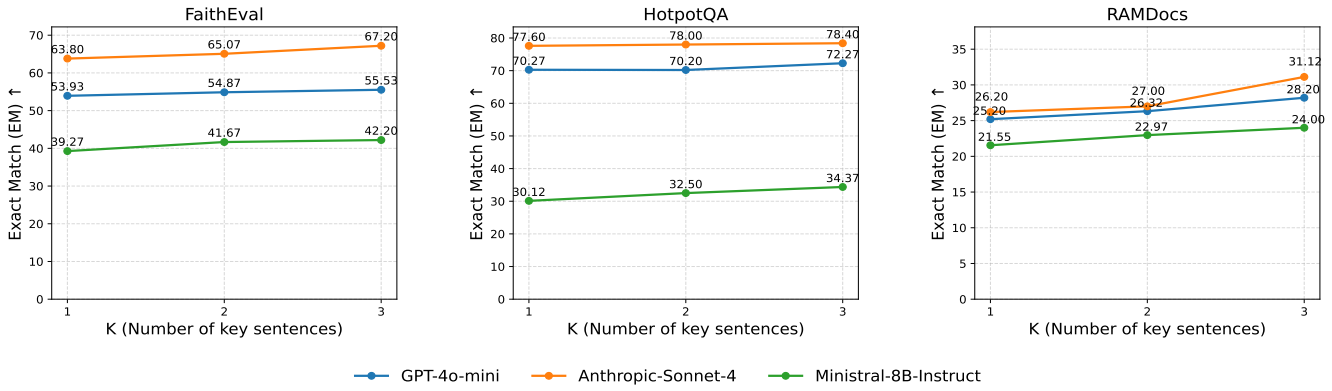

Figure 2: Ablation on the number of key sentences (K). Metrics are EM↑ for FaithEval, HotpotQA, and RAMDocs. Each subplot shows performance trends across models (GPT-4o-mini, Anthropic-Sonnet-4, Ministral-8B-Instruct) as K increases.

Table 2: Effect of NLI model and Agent debate across backbones. Metrics are EM↑ for FaithEval/RAMDocs and HotpotQA.

	FaithEval (Inconsistent)			HotpotQA			RAMDocs		
Variant	GPT-4o-mini	Sonnet-4	Ministral 8B-Inst.	GPT-4o-mini	Sonnet-4	Ministral 8B-Inst.	GPT-4o-mini	Sonnet-4	Ministral 8B-Inst.
NLI 1 (base)	55.53	67.20	42.20	**72.27**	**78.40**	**34.37**	**28.20**	**31.12**	**24.00**
NLI 2	**55.80**	**68.13**	**43.80**	70.73	77.13	32.74	27.00	30.40	22.46
Agent	53.87	65.32	40.84	68.37	75.23	31.67	23.60	28.35	19.23

aggregation, and generation stages. When this simple principle operates on a sentence–relation graph, the reasoning process becomes clearer even in long and complex contexts, and conflicting signals are naturally suppressed.

Experimentally, EG-RAG demonstrated consistent improvements across three heterogeneous backbones (GPT-4o-mini, Anthropic-Sonnet-4, Ministral-8B-Instruct) and three benchmarks (FaithEval–Inconsistent, HotpotQA, RAMDocs), achieving strong performance in conflict-verification and multi-document reasoning tasks while remaining robust under noisy and misinformation-rich conditions. Performance remained stable even when the number of key sentences was limited to 1–3, confirming high efficiency relative to cost. Moreover, comparisons between NLI classifiers and Agent-based relation modeling revealed that enhanced NLI classifiers excel in adversarial settings, whereas conservative baselines are more stable in complex retrieval environments. In summary, EG-RAG maintains structural evidence throughout selection, aggregation, and generation, improving both factual faithfulness and multi-document reasoning.

Several implications arise from this study. First, sentence-level structuring significantly improves interpretability compared to simple concatenation, allowing intuitive tracing of where supporting or contradictory signals originate. Second, EG-RAG operates as a model-agnostic procedure compatible with both large-scale API models and lightweight open-source models, reducing deployment risk when transitioning from research to production. Third, the benefits of structural modeling interact with retrieval quality: support clusters consolidate reasoning paths when retrieval is strong, while contradiction clusters act as safeguards when retrieval is noisy.

Future research directions include: (1) adaptive graph construction that dynamically adjusts key sentence counts, edge thresholds, and cluster weighting according to query and domain characteristics; (2) hybrid reasoning frameworks where the evidence graph serves as a shared board in an explicit multi-agent architecture, with cluster-level agents negotiating only unresolved conflicts to combine structural efficiency with deliberative depth; and (3) integration of source reliability and domain priors into nodes and edges for more dynamic weighting of contradictory signals, further enhancing robustness against misinformation and noise.

In conclusion, EG-RAG demonstrates that explicitly structuring sentence-level relations into a unified evidence graph is a simple yet effective principle for achieving reliable reasoning in complex retrieval settings, and we expect this direction to generalize to broader knowledge-intensive tasks.

ACKNOWLEDGMENTS

This work was the result of project supported by KT (Korea Telecom). This work was also supported by the IITP grant funded by the Korean government (MSIT) under IITP-2026-RS-2020-II201821 (30%), RS-2024-00392332 (30%), RS-2019-II190421 (10%), and the National Research Foundation of Korea (NRF) grant funded by the Korea government (MSIT) (RS-2024-00343255, 30%).

REFERENCES

[1] Anthropic. 2025. Claude 4.0 Sonnet Model Card. https://www.anthropic.com/claude/sonnet

[2] Akari Asai, Zeqiu Wu, Yizhong Wang, Avirup Sil, and Hannaneh Hajishirzi. 2024. Self-RAG: Learning to Retrieve, Generate, and Critique through Self-Reflection. In *Proceedings of the Twelfth International Conference on Learning Representations (ICLR 2024).*

[3] Samuel R. Bowman, Gabor Angeli, Christopher Potts, and Christopher D. Manning. 2015. A Large Annotated Corpus for Learning Natural Language Inference. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics, Lisbon, Portugal, 632–642.

[4] Wenqi Fan, Yujuan Ding, Liangbo Ning, Shijie Wang, Hengyun Li, Dawei Yin, Tat-Seng Chua, and Qing Li. 2024. A Survey on RAG Meeting LLMs: Towards Retrieval-Augmented Large Language Models. In *KDD*. ACM, 6491–6501.

[5] Kelvin Guu, Kenton Lee, Zora Tung, Panupong Pasupat, and Ming-Wei Chang. 2020. REALM: Retrieval-Augmented Language Model Pre-Training. In *ICML (PMLR, Vol. 119)*. PMLR, 3929–3938.

[6] Pengcheng He, Jianfeng Gao, and Weizhu Chen. 2021. Debertav3: Improving deberta using electra-style pre-training with gradient-disentangled embedding sharing. *arXiv preprint arXiv:2111.09543* (2021).

[7] Yuntong Hu, Zhihan Lei, Zheng Zhang, Bo Pan, Chen Ling, and Liang Zhao. 2025. GRAG: Graph Retrieval-Augmented Generation. In *Findings of the Association for Computational Linguistics: NAACL 2025*. Association for Computational Linguistics, Albuquerque, New Mexico, 4145–4157.

[8] Gautier Izacard and Edouard Grave. 2021. Leveraging Passage Retrieval with Generative Models for Open Domain Question Answering. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume*. Association for Computational Linguistics, Online, 874–880.

[9] Vladimir Karpukhin, Barlas Oguz, Sewon Min, Patrick Lewis, Ledell Wu, Sergey Edunov, Danqi Chen, and Wen-tau Yih. 2020. Dense Passage Retrieval for Open-Domain Question Answering. In *EMNLP*. ACL, 6769–6781.

[10] 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. 2020. Retrieval-Augmented Generation for Knowledge-Intensive NLP Tasks. In *Advances in Neural Information Processing Systems*, Vol. 33. 9459–9474.

[11] Weitao Li, Junkai Li, Weizhi Ma, and Yang Liu. 2024. Citation-Enhanced Generation for LLM-based Chatbots. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. Association for Computational Linguistics, Bangkok, Thailand, 1451–1466.

[12] Teng Lin. 2025. Structured Retrieval-Augmented Generation for Multi-Entity Question Answering over Heterogeneous Sources. , 253–258 pages.

[13] Hao Liu, Zhengren Wang, Xi Chen, Zhiyu Li, Feiyu Xiong, Qinhan Yu, and Wentao Zhang. 2025. HopRAG: Multi-Hop Reasoning for Logic-Aware Retrieval-Augmented Generation. In *Findings of the Association for Computational Linguistics: ACL 2025*. Association for Computational Linguistics, Vienna, Austria, 1897–1913.

[14] Nelson F. Liu, Kevin Lin, John Hewitt, Ashwin Paranjape, Michele Bevilacqua, Fabio Petroni, and Percy Liang. 2024. Lost in the Middle: How Language Models Use Long Contexts. *Transactions of the Association for Computational Linguistics* 12 (2024), 157–173.

[15] Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. Roberta: A robustly optimized bert pretraining approach. *arXiv preprint arXiv:1907.11692* (2019).

[16] Shayne Longpre, Kartik Perisetla, Anthony Chen, Nikhil Ramesh, Chris DuBois, and Sameer Singh. 2021. Entity-Based Knowledge Conflicts in Question Answering. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics, Online and Punta Cana, Dominican Republic, 7052–7063.

[17] Aman Madaan, Niket Tandon, Prakhar Gupta, Skyler Hallinan, Luyu Gao, Sarah Wiegreffe, Uri Alon, Nouha Dziri, Shrimai Prabhumoye, Yiming Yang, et al. 2023. Self-refine: Iterative refinement with self-feedback. *Advances in Neural Information Processing Systems* 36 (2023), 46534–46594.

[18] Yifei Ming, Senthil Purushwalkam, Shrey Pandit, Zixuan Ke, Xuan-Phi Nguyen, Caiming Xiong, and Shafiq Joty. 2025. FaithEval: Can Your Language Model Stay Faithful to Context, Even If "The Moon is Made of Marshmallows". In *Proceedings of the International Conference on Learning Representations (ICLR 2025)*.

[19] Mistral AI. 2024. Ministral-8B-Instruct-2410 (Model Card). https://huggingface.co/mistralai/Ministral-8B-Instruct-2410 Hugging Face model card for Ministral-8B-Instruct-2410.

[20] OpenAI. 2024. GPT-4o mini: Advancing Cost-Efficient Intelligence. https://openai.com/index/gpt-4o-mini-advancing-cost-efficient-intelligence/

[21] Jason Phang, Angelica Chen, William Huang, and Samuel R. Bowman. 2022. Adversarially Constructed Evaluation Sets Are More Challenging, but May Not Be Fair. In *Proceedings of the First Workshop on Dynamic Adversarial Data Collection*. Association for Computational Linguistics, Seattle, WA, 62–62.

[22] James Thorne, Andreas Vlachos, Christos Christodoulopoulos, and Arpit Mittal. 2018. FEVER: a Large-scale Dataset for Fact Extraction and Verification. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*. Association for Computational Linguistics, New Orleans, Louisiana, 809–819.

[23] Fei Wang, Xingchen Wan, Ruoxi Sun, Jiefeng Chen, and Sercan O. Arik. 2025. Astute RAG: Overcoming Imperfect Retrieval Augmentation and Knowledge Conflicts for Large Language Models. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. Association for Computational Linguistics, Vienna, Austria, 30553–30571.

[24] Han Wang, Archiki Prasad, Elias Stengel-Eskin, and Mohit Bansal. 2025. Retrieval-Augmented Generation with Conflicting Evidence. In *Second Conference on Language Modeling*.

[25] Weikang Wang, Guanhua Chen, Hanqing Wang, Yue Han, and Yun Chen. 2022. Multilingual Sentence Transformer as A Multilingual Word Aligner. In *Findings of the Association for Computational Linguistics: EMNLP 2022*. Association for Computational Linguistics, Abu Dhabi, United Arab Emirates, 2952–2963.

[26] Adina Williams, Nikita Nangia, and Samuel Bowman. 2018. A Broad-Coverage Challenge Corpus for Sentence Understanding through Inference. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*. Association for Computational Linguistics, New Orleans, Louisiana, 1112–1122. https://doi.org/10.18653/v1/N18-1101

[27] Adina Williams, Tristan Thrush, and Douwe Kiela. 2022. ANLIzing the Adversarial Natural Language Inference Dataset. In *Proceedings of the Society for Computation in Linguistics 2022*, Allyson Ettinger, Tim Hunter, and Brandon Prickett (Eds.). Association for Computational Linguistics, online, 23–54.

[28] Zhenyu Wu, Qingkai Zeng, Zhihan Zhang, Zhaoxuan Tan, Chao Shen, and Meng Jiang. 2024. Large Language Models Can Self-Correct with Key Condition Verification. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics, Miami, Florida, USA, 12846–12867.

[29] Zhilin Yang, Peng Qi, Saizheng Zhang, Yoshua Bengio, William W. Cohen, Ruslan Salakhutdinov, and Christopher D. Manning. 2018. HotpotQA: A Dataset for Diverse, Explainable Multi-hop Question Answering. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics, Brussels, Belgium, 2369–2380.

[30] Qinggang Zhang, Zhishang Xiang, Yilin Xiao, Le Wang, Junhui Li, Xinrun Wang, and Jinsong Su. 2025. FaithfulRAG: Fact-Level Conflict Modeling for Context-Faithful Retrieval-Augmented Generation. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (ACL 2025)*. Vienna, Austria, 21863–21882.

[31] Jie Zhou, Xu Han, Cheng Yang, Zhiyuan Liu, Lifeng Wang, Changcheng Li, and Maosong Sun. 2019. GEAR: Graph-based Evidence Aggregating and Reasoning for Fact Verification. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*. Association for Computational Linguistics, Florence, Italy, 892–901.