NEURAL INFORMATION PROCESSING SYSTEMS

Learning Minimal Sufficient Evidence Graphs for GraphRAG via Nash-Guided Optimization

Hao Wu

Supervisor: Dr. Jack Yang

Preliminaries

Instead of reasoning over an unordered set of text fragments (as instandard RAG), GraphRAG operates on structured evidence composed of **nodes** and **edges.**

Offline GraphRAG

A corpus-level graph G is constructed in advance and remains fixed during retrieval.

$$P(a \mid q, G) = \sum_{G_q \subseteq G} P(a \mid q, G_q) P(G_q \mid q, G)$$

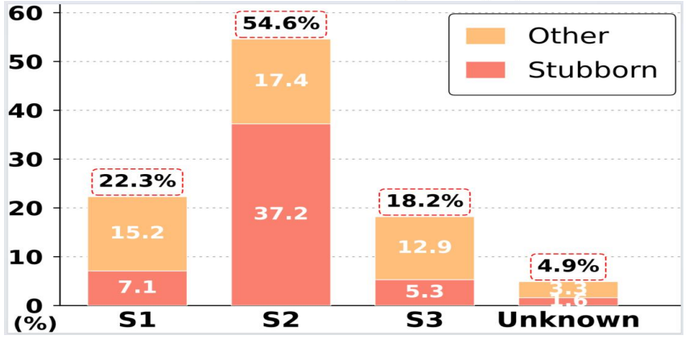
Online GraphRAG

The evidence graph is incrementally constructed and refined for the target query.

$$P(a \mid q) = \sum_{G_q \in \mathcal{G}(q)} P(a \mid q, G_q) P(G_q \mid q)$$

Query-Conditioned Knowledge Conflict

S1) Retrieval noise
S2) Counterfactual fact
S3) Semantic ambiguity



Prevalence and diversity of query conditioned knowledge conflict in GraphRAG : ConFiQA-MC

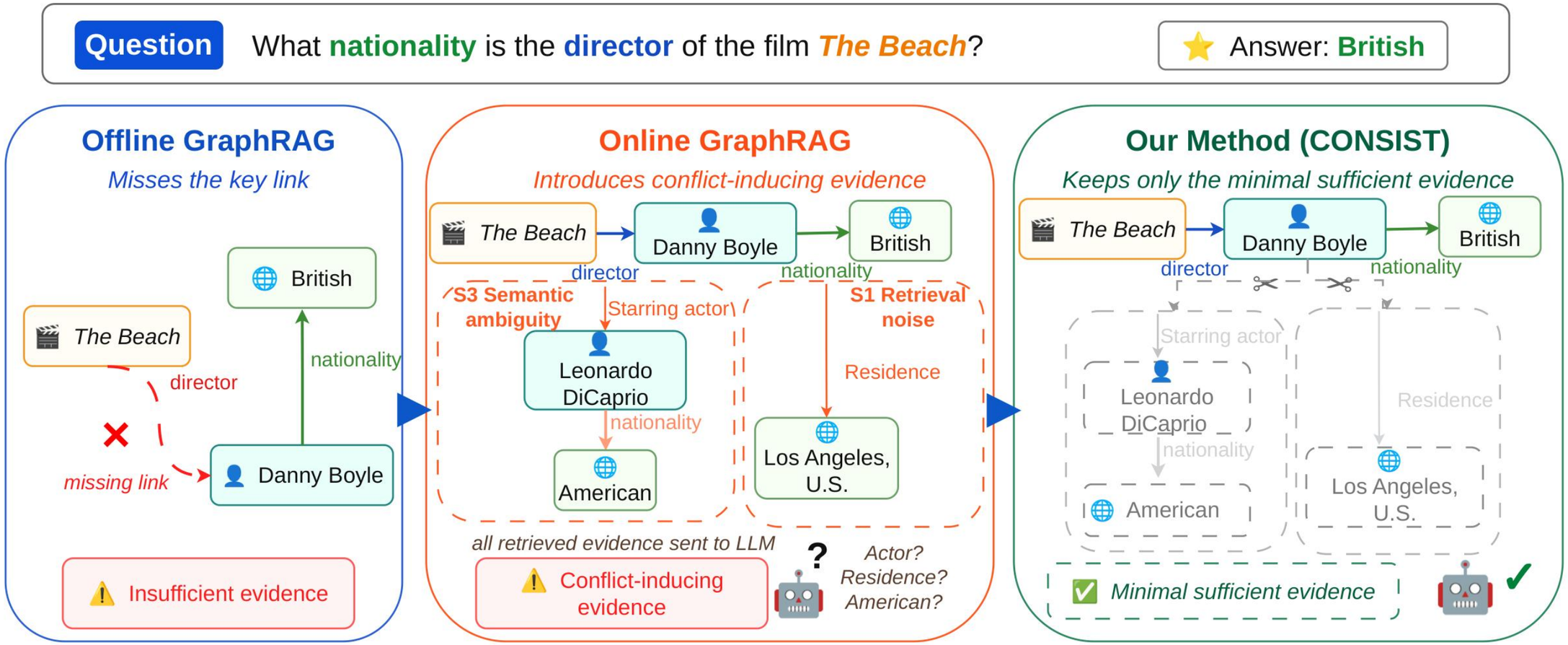
Nash Bargaining Maximizing joint gains relative to a disagreement point.

$$u^* = \arg\max_{u_i \in \mathcal{U}} \sum_{i=1}^{K} \log(u_i - d_i), \qquad \text{s.t. } \forall i: \; u_i > d_i.$$

Gap

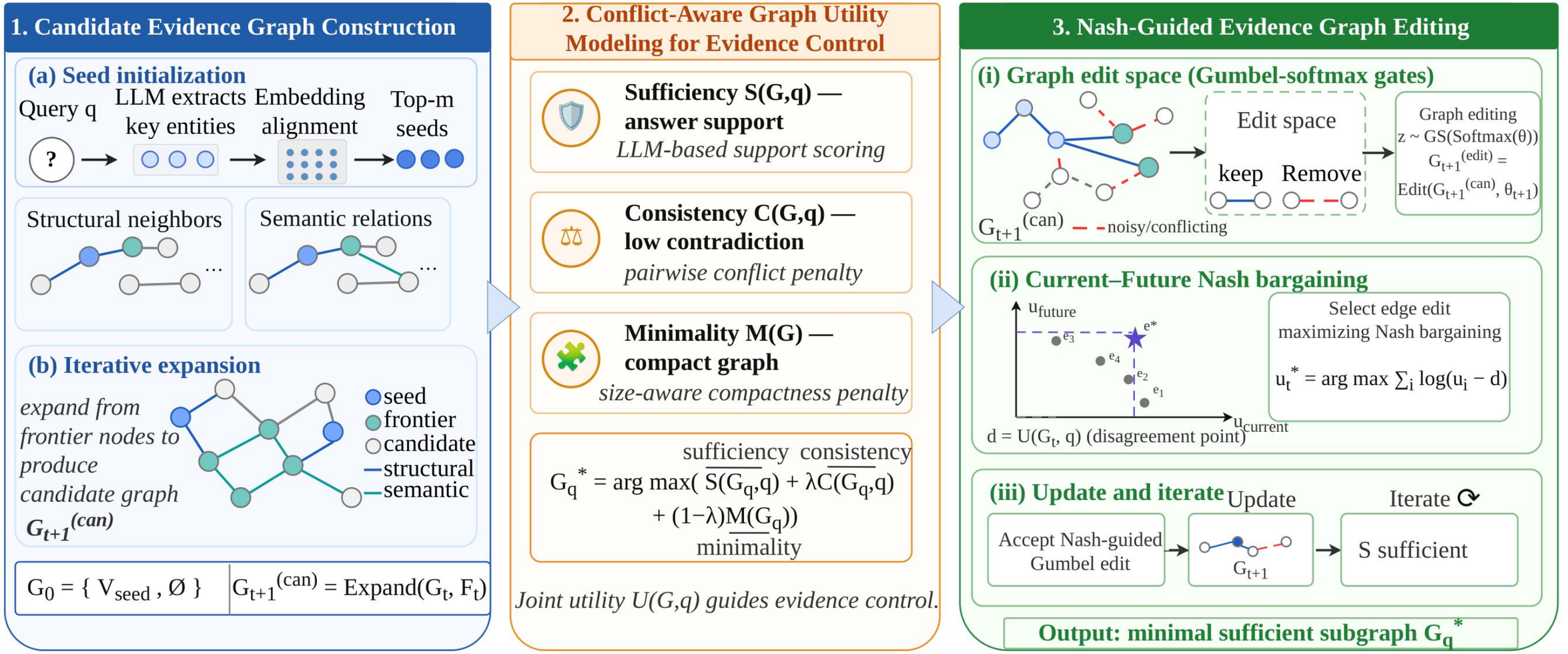
Motivation

How can GraphRAG construct an evidence graph that is **sufficient for answering a query** while remaining **compact and conflict-free** ?



METHOD

CONSIST: Exploration and Editing



METHOD

Problem Formulation

For a query q and a set of document nodes V with an initial edge set E, let $\mathcal{G}(q)$ denote the query-specific graph space reachable through **edge expansion or removal** over V, where edges may originate from E or be dynamically introduced beyond E.

$$G_q^* = \arg\max_{G_q \in \mathcal{G}(q)} S(G_q, q) + \lambda C(G_q, q) + (1 - \lambda) M(G_q)$$

Sufficiency G_q provides evidence that supports answering q.

Consistency G_q avoids contradictory or incompatible knowledge.

Minimality G_q remains compact, minimizing duplicated or unnecessary evidence.

λ is a trade-off weight between consistency and minimality.

CORE STEP 1

Candidate Evidence Graph Construction

Seed initialization. Extract query entities, align them with document nodes, and select top-m LLM-ranked seeds.

$$G_0 = \{V_{\text{seed}}, \varnothing\}, \qquad F_0 = V_{\text{seed}}$$

Candidate expansion. Expand **structural neighbors** and **semantic relations.**

$$V_{t+1} = V_t \cup \bigcup_{v \in F_t} \left(N_{\text{str}}(v) \cup N_{\text{sem}}(v)\right),$$

$$E_{t+1} = E_t \cup \{(v, v_{\text{str,sem}}) \mid v \in F_t,\ v_{\text{str,sem}} \in N_{\text{str}}(v) \cup N_{\text{sem}}(v)\}.$$

If no prior edges exist, candidate expansion relies solely on semantic relations.

CORE STEP 2

Conflict-Aware Graph Utility

Sufficiency

$$S(G, q) = \mathrm{LLM}_{\mathrm{sup}}(G, q)$$

The answer-support scorer rewards evidence useful for answering the query.

Consistency

$$C(G, q) = 1 - \mathrm{Avg}_{(e_i, e_j) \in E(G)} \mathrm{LLM}_{\mathrm{pair}}\big(q, (e_i, e_j)\big)$$

$\mathrm{LLM}_{\mathrm{pair}}$ scores conflict between evidence edges.

Minimality

$$M(G) = \frac{|E(G')|}{|E(G)|}$$

G' denotes the graph before expansion. The edge-count ratio acts as a **compactness proxy** .

$$U(G, q) \quad = \quad S(G, q) \; + \; \lambda C(G, q) \; + \; (1 \; - \; \lambda) M(G)$$

CORE STEP 3

Nash-Guided Graph Editing

Each gate determines whether a candidate edge is retained. A **Gumbel-based relaxation** enables gradient-based editing.

$$\mathrm{GS}(\theta) \;=\; \sigma\left(\frac{\log\epsilon - \log(1-\epsilon) + \theta}{\Delta}\right), \qquad \epsilon \;\sim\; \mathcal{U}(0,1)$$

$$G_{t+1}^{(\mathrm{edit})} = \mathrm{Edit}\left(G_{t+1}^{(\mathrm{can})}, \theta_{t+1}\right)$$

$$\theta_{t+1}^{e} = \begin{cases} \theta_t^e, & e \in E_t, \\ 1, & \text{otherwise.} \end{cases}$$

Nodes are removed when all incident edges are pruned. New edges are initialized at **1**, giving them a temporary retention preference before utility-based editing.

CORE STEP 3

Nash-Guided Evidence Graph Editing

Current and future players. Immediate improvement can overlook conflicts introduced by later expansion. The rollout uses H further expansion steps **without additional editing** .

$$u_{\text{cur}} = \mathbb{E}\left[U\left(G_{t+1}^{(\text{edit})}(\theta), q\right)\right], \quad u_{\text{fut}} = \mathbb{E}\left[U\left(G_{t+1+H}^{(\text{edit})}(\theta), q\right)\right].$$

$$u_{t+1}^{*} = \arg \max_{u_i \in \mathcal{U}_{t+1}(\theta)} \sum_{i \in \{\text{cur},\text{fut}\}} \log(u_i - d_t), \quad \text{s.t. } \forall i,\ u_i > d_t.$$

$d_t = \mathbb{E}[U(G_t, q)]$: **both players must improve over the unchanged graph** .

If no such edit exists, CONSIST keeps the current graph unchanged.

Monte Carlo sampling. Stop when $S(G, q) \geq \tau_{\text{stop}}$, then generate $\hat{a} = \text{LLM}(G_q^*, q)$.

Experiments: Multi-hop QA

Setup. We evaluate EM/F1 on 2Wiki, HotpotQA, and MuSiQue.

Method	2Wiki		HotpotQA		MuSiQue		Avg.	
	EM(↑)	F1(↑)	EM(↑)	F1(↑)	EM(↑)	F1(↑)	EM(↑)	F1(↑)
DeepSeek-V3	31.2	36.5	29.4	40.2	7.8	16.4	22.8	31.0
+ RAPTOR [32]	46.7	54.8	47.2	63.1	18.8	28.6	37.6	48.8
+ TOG [23]	32.8	37.3	24.1	29.2	5.3	7.0	20.7	24.5
+ G-Retriever [8]	36.2	45.5	36.9	49.3	9.2	16.9	27.4	37.2
+ LightRAG [6]	30.0	36.4	38.4	49.8	10.4	17.1	26.3	34.4
+ GraphRAG [5]	31.8	37.9	45.0	56.9	13.8	20.6	30.2	38.5
+ HippoRAG [7]	57.8	68.4	49.8	64.7	19.0	29.8	42.2	54.3
+ Relink [11]	62.8	72.2	55.8	70.4	25.2	37.0	47.9	59.9
+ CONSIST	**65.3**	**73.9**	**62.6**	**77.8**	**43.5**	**55.9**	**57.1**	**69.2**
Std.	±1.0	±0.9	±1.3	±1.2	±0.7	±1.1	±1.0	±1.1
GPT-4o	29.2	35.8	33.0	42.4	10.6	21.4	24.3	33.2
+ SiReRAG [49]	59.6	67.9	61.7	76.5	40.5	53.1	53.9	65.8
+ TCR-QF [10]	59.8	68.0	55.8	70.8	30.3	43.2	48.6	60.7
+ HopRAG [20]	61.1	68.3	62.0	76.1	42.2	54.9	55.1	66.4
+ CONSIST	**67.2**	**75.7**	**64.3**	**79.0**	**44.1**	**56.8**	**58.5**	**70.5**
Std.	±1.1	±0.8	±0.9	±0.8	±0.5	±1.0	±0.8	±0.9

CONSIST achieves the **best EM/F1** on 2Wiki, HotpotQA, and MuSiQue. DeepSeek-V3 average EM improves from ReLink's 47.9 to **57.1**.

Experiments: Conflict-oriented QA

Setup. FaithEval and three ConFiQA settings.

Method	FaithEval		ConFiQA (MC)		ConFiQA (MR)		ConFiQA (QA)		Avg.	
	EM(↑)	F1(↑)	EM(↑)	F1(↑)	EM(↑)	F1(↑)	EM(↑)	F1(↑)	EM(↑)	F1(↑)
LLaMA-3-8B	6.0	27.7	2.1	5.0	1.9	6.1	1.3	6.1	2.8	11.2
+ OPIN(INSTR) [52]	15.1	34.9	57.3	67.4	54.0	65.9	67.4	76.9	48.5	61.3
+ CAD [34]	42.7	59.4	11.4	16.0	31.3	40.0	38.1	48.3	30.9	40.9
+ COIECD [47]	41.3	56.1	24.0	28.5	43.3	50.9	60.1	67.1	42.2	50.7
+ ADACAD [37]	47.7	61.4	32.5	43.3	42.5	46.7	48.1	55.7	42.7	51.8
+ KRE [46]	12.1	59.1	59.8	68.2	58.9	68.7	74.7	84.0	51.4	70.0
+ FAITHFULRAG [50]	49.1	62.4	61.2	70.3	53.1	62.4	65.2	70.4	57.2	66.4
+ CONSIST	**55.7**	**68.5**	**65.1**	**73.9**	**63.8**	**72.7**	**77.6**	**87.3**	**65.6**	**75.6**
Std.	±1.2	±1.5	±0.7	±0.8	±0.6	±0.6	±0.3	±0.2	±0.7	±0.8

In multi-hop QA, it preserves necessary intermediate links; in conflict-heavy settings, it suppresses **inconsistent or ambiguous paths** that could mislead reasoning.

EXPERIMENTS

Model Breakdown

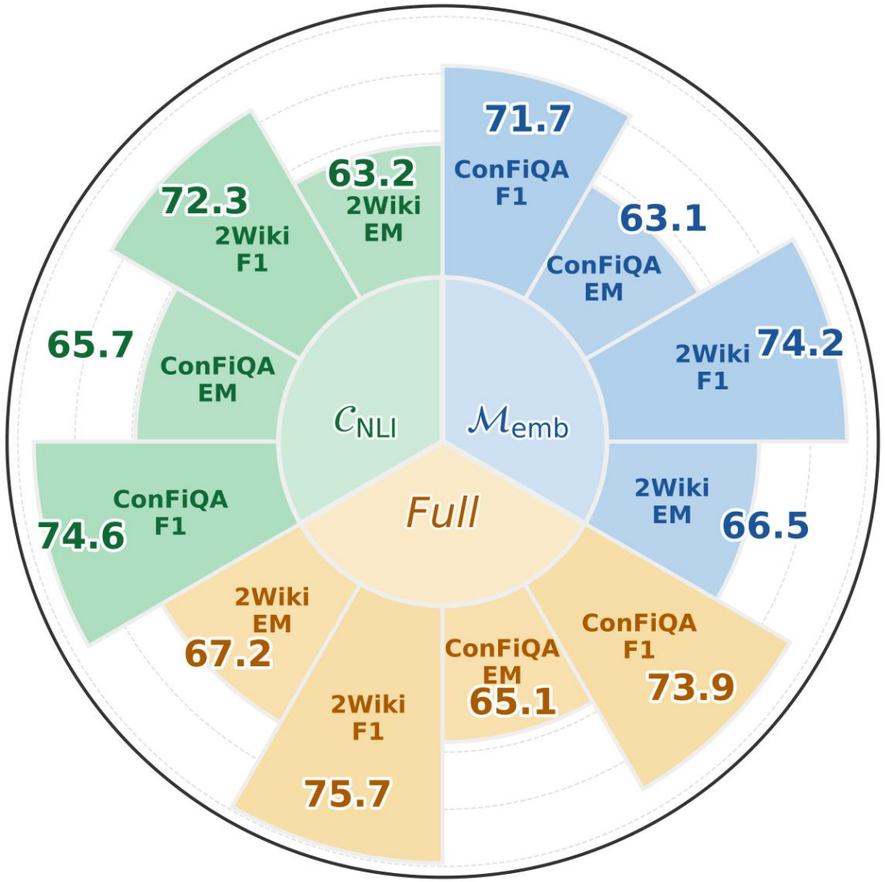
	2Wiki		ConFiQA-MC	
Variant	EM	F1	EM	F1
+EXP	56.2	65.1	49.4	55.6
+GS-LLM	59.4	68.5	54.9	60.4
+GS-U	63.9	72.8	61.7	67.3
+GS-SUM	64.6	73.4	63.2	70.1
Full	**67.2**	**75.7**	**65.1**	**73.9**

EXP: expansion-only
GS-LLM: LLM-derived heuristic
GS-U: current-player utility
GS-SUM: sum of current and future utilities
Full: Nash bargaining with our proposed disagreement point

Nash coordination further gains 2.3 F1 points on 2Wiki and 3.8 on ConFiQA-MC over direct utility summation.

EXPERIMENTS

Utility Ablation



NLI-based Consistency

Replaces the LLM-based consistency with a natural language inference-based contradiction detector.

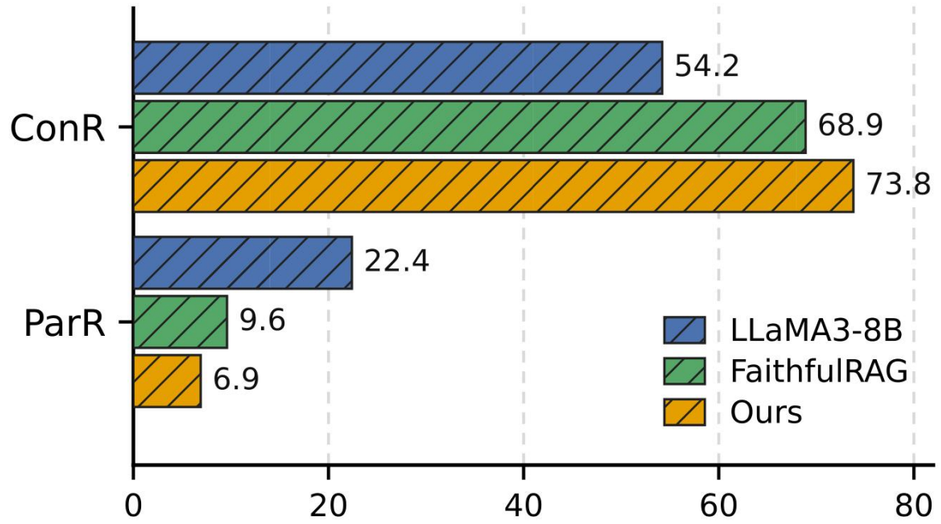
Embedding-based Minimality

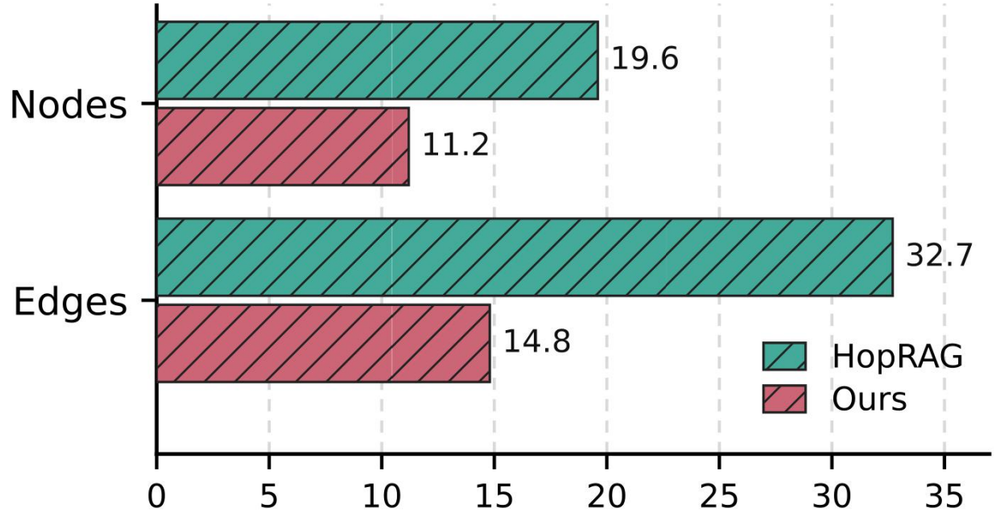
Replaces the size-based minimality with an embedding-based redundancy to count nodes if their cosine similarity is greater than 0.75.

EXPERIMENTS

Graph Quality: Consistency and Minimality

ConR: context alignment. **ParR** : parametric-knowledge alignment. Higher ConR together with lower ParR indicates better conflict resolution.



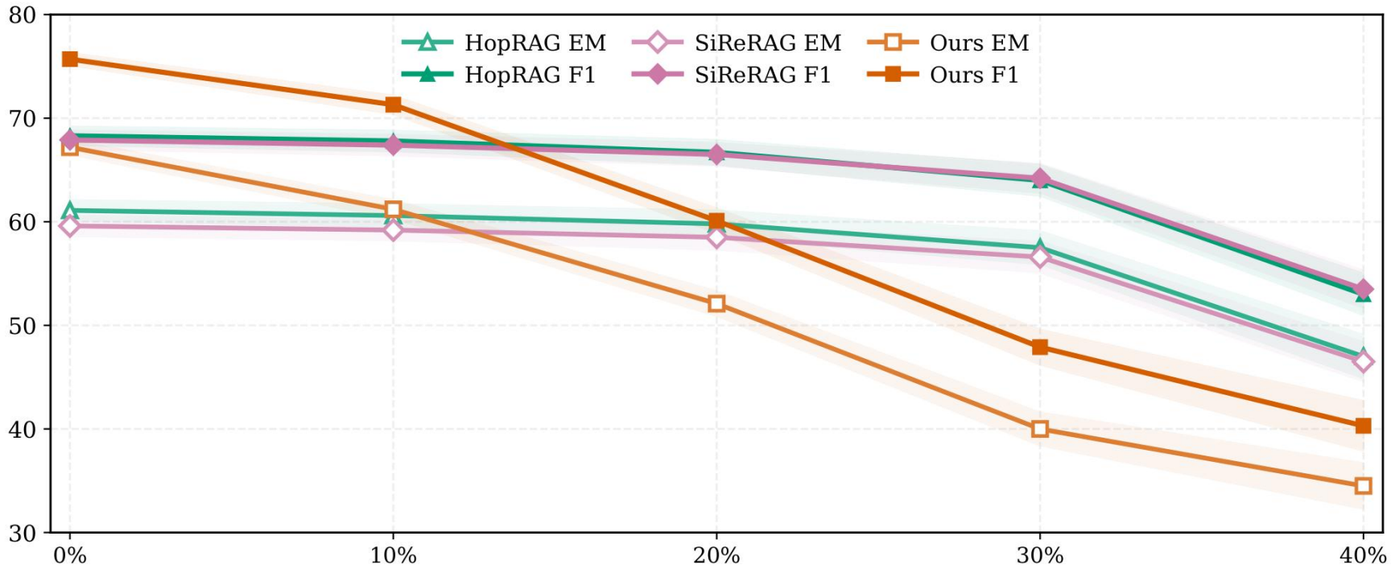


On 2Wiki, CONSIST retains **11.2 nodes and 14.8 edges** , versus HopRAG's 19.6 and 32.7: **42.9% fewer nodes** and **54.7% fewer edges** .

FURTHER ANALYSIS

Model Robustness via Evidence Pruning

Setup. Randomly remove $\xi\%$ of edges from the final evidence graph, with $\xi \in \{0\%, 10\%, 20\%, 30\%, 40\%\}$, and evaluate the resulting performance.



Result. CONSIST degrades **more sharply** than baseline methods.

Explanation. A higher proportion of **task-critical edges**, with fewer duplicated reasoning paths.

The evidence selected by CONSIST is **more compact and less redundant**.

FURTHER ANALYSIS

Time Complexity Analysis

Each iteration introduces at most b new edges: $|E_t| \leq |E_0| + tb$. When the initial graph size is treated as constant, $|E_t| = \mathcal{O}(tb)$.

Cost at iteration t

$$\boxed{\mathcal{O}((1+H)bC_{\text{exp}} + tb + C_{\text{LLM}})}$$

C_{exp}: cost of expanding one candidate edge.

C_{LLM}: cost of one LLM-based operation.

The additional editing cost grows **linearly with the accumulated expansion budget** .

$$\boxed{\textsc{ReLink}: \quad \mathcal{O}(b(C_{\text{exp}} + C_{\text{LLM}}) + b\log b)\,.}$$

$$\boxed{\textsc{HopRAG}: \quad \mathcal{O}\big(bC_{\text{exp}} + b^2 C_{\text{LLM}}\big)\,.}$$

$$\boxed{\textsc{ToG-3}: \quad \mathcal{O}(Rb(C_{\text{exp}} + C_{\text{LLM}}))\,.}$$

Method	Avg. time (s)
RELINK	12.62
HopRAG	17.52
ToG-3	15.18
CONSIST	**12.80**

Runtime comparable to RELINK and more efficient than HopRAG and ToG-3.

NASH-GUIDED OPTIMIZATION

Proposition : Continuity of Expected Utility

Let $\mathcal{G}$ be a **finite set of graphs** and $h : \mathcal{G} \mapsto \mathbb{R}$. Let g be a random graph generated by Gumbel-softmax edge-retention probabilities. Assume $\Delta > 0$.

$u(\theta) = \mathbb{E}[h(g) \mid \theta]$ is **continuous in** θ.

Each $p_i(\theta)$ is continuous. Under independent Bernoulli edge-retention decisions conditional on θ:

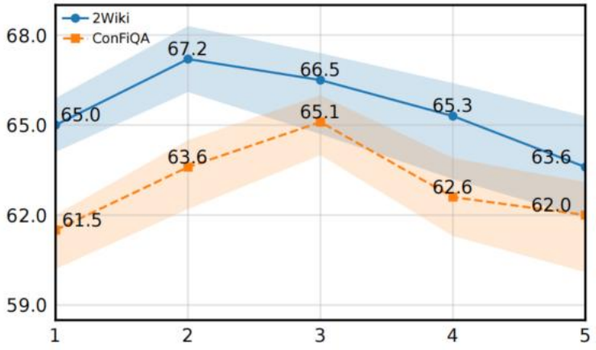
$$P(g = g_x \mid \theta) = \prod_{i=1}^{m} p_i(\theta)^{x_i}(1 - p_i(\theta))^{1-x_i}, \quad u(\theta) = \sum_{g \in \mathcal{G}} h(g)P(g \mid \theta).$$

A finite product of continuous functions is continuous. $u(\theta)$ is a finite linear combination of continuous functions.

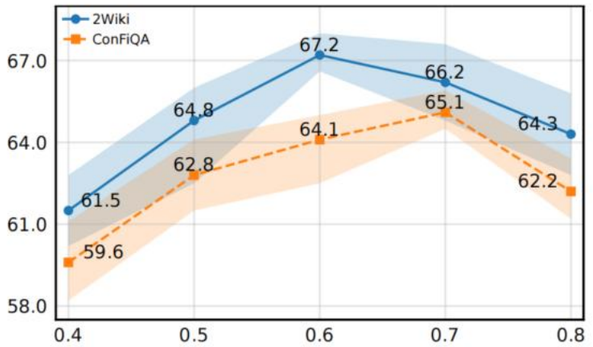
Replacing $h(g)$ by $h(f(g))$: u_{fut} **is also continuous in** θ.

BACKUP

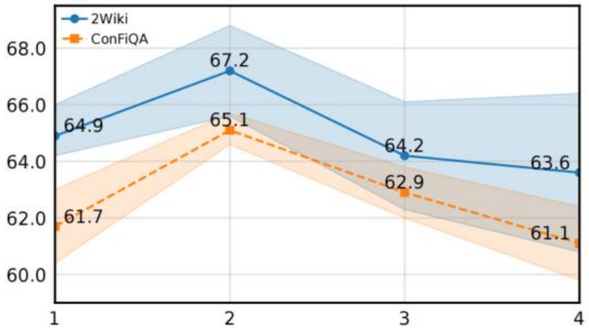
Sensitivity Analysis



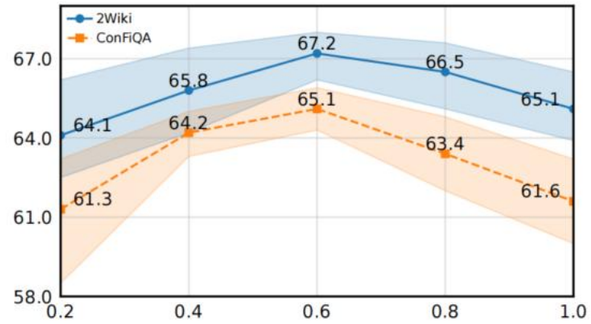
(a) Number of seeds (m).



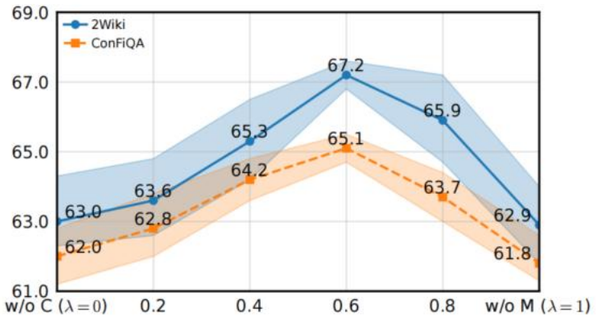
(b) Semantic expansion threshold τ_{sem}.



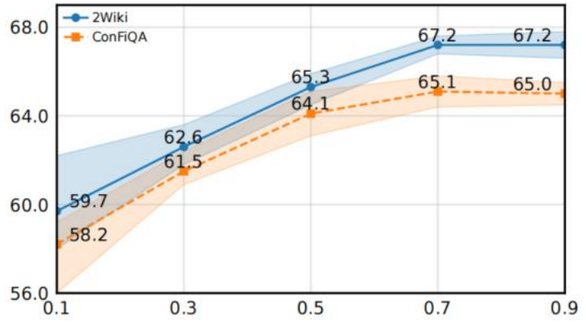
(a) Rollout step H.



(b) Gumbel-Softmax temperature Δ.



(a) Utility hyperparameter λ.



(b) Termination threshold τ_{stop}.

Conclusion

CONSIST explicitly regulates graph construction by jointly modeling **sufficiency, consistency, and minimality**, and refines candidate graphs through Gumbel-based edge editing guided by **current–future Nash bargaining**.

Experiments show that CONSIST consistently improves **answer accuracy** while producing substantially more **compact evidence graphs**.