SUNAR: Semantic Uncertainty based Neighborhood Aware Retrieval for Complex QA

Venktesh V[*]
TU Delft
v.viswanathan-1
@tudelft.nl

Mandeep Rathee[*]
L3S Research Center
rathee
@l3s.de

Avishek Anand
TU Delft
avishek.anand
@tudelft.nl

Abstract

Complex question-answering (QA) systems face significant challenges in retrieving and reasoning over information that addresses multi-faceted queries. While large language models (LLMs) have advanced the reasoning capabilities of these systems, the **bounded-recall** problem persists, where procuring all relevant documents in first-stage retrieval remains a challenge. Missing pertinent documents at this stage leads to performance degradation that cannot be remedied in later stages, especially given the limited context windows of LLMs which necessitate high recall at smaller retrieval depths. In this paper, we introduce SUNAR, a novel approach that leverages LLMs to guide a Neighborhood Aware Retrieval process. SUNAR iteratively explores a neighborhood graph of documents, dynamically promoting or penalizing documents based on uncertainty estimates from interim LLM-generated answer candidates. We validate our approach through extensive experiments on two complex QA datasets. Our results show that SUNAR significantly outperforms existing retrieve-and-reason baselines, achieving up to a **31.84%** improvement in performance over existing state-of-the-art methods for complex QA.

 github.com/VenkteshV/SUNAR

1 Introduction

Open-domain complex question answering (CQA) has emerged as a critical challenge in natural language processing, demanding systems to comprehend, reason, and synthesize information over multiple queries and from multiple sources. While effective for simpler queries, traditional retrieve-and-reason pipelines struggle with the multifaceted nature of complex QA tasks. Complex QA systems typically involve multiple stages: query understanding, retrieval of relevant documents, and a
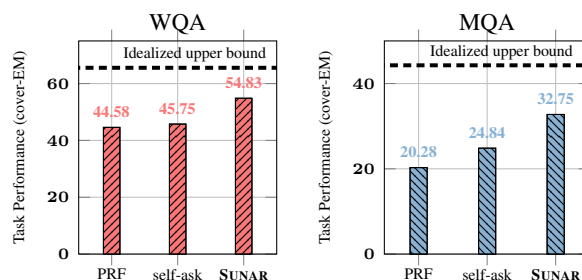

Figure 1: The graph represents the comparison general decompose-retrieve paradigm for complex QA (SELF-ASK), and the core approach of this work SUNAR in open-domain setup for complex QA on 2WikiMulti-hopQA (WQA) (left) and MusiqueQA (MQA) (right).

reasoning phase that may or may not leverage large language models (LLMs). However, the effectiveness of these systems is typically bounded by the recall of the retrieval stage. Specifically, if relevant documents are missed during this stage, they cannot be incorporated later, leading to a decline in overall system performance.

This *bounded-recall problem* is more prominent in complex question answering where the initial question comprises multiple sub-questions that necessitate a retrieval for each sub-question (Ho et al., 2020; Trivedi et al., 2022). Failing to retrieve relevant documents to answer a sub-question, causes cascading failures that result in suboptimal performance of CQA systems. Furthermore, the limited context windows of LLMs necessitate a focus on recall at smaller retrieval depths.

Existing retrieval literature primarily focuses on improving precision at higher retrieval depths by either employing re-rankers (Nogueira et al., 2019), or using improving the LLM-reasoning by using self-correction mechanisms and decomposition strategies (Press et al., 2023; Khattab et al., 2023; Asai et al., 2024). While these efforts led to notable improvements in performance in comparison to the standard open-domain QA baselines, the

---

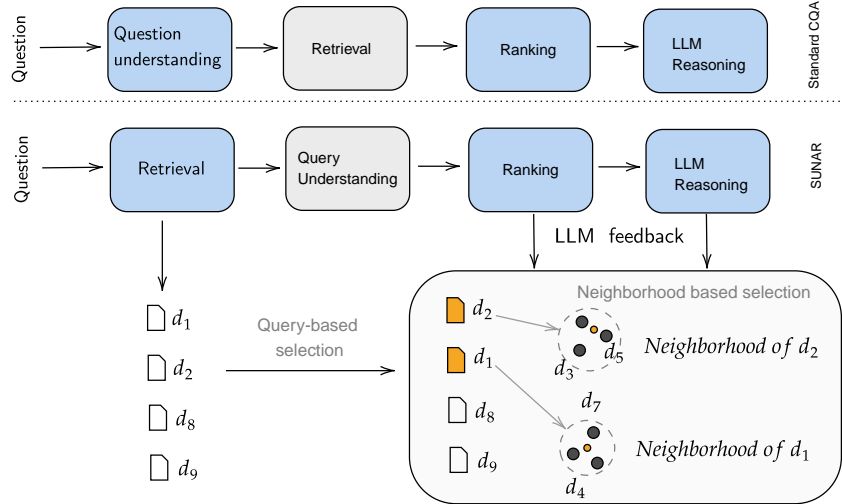[*]Both authors contributed equally to this work

Figure 2: Overview of SUNAR with Neighborhood Aware Retrieval (NAR) and LLM based feedback.

recall problem still persists as shown in Figure 1.

In this paper, we propose a novel method to bridge the recall gap by employing LLM feedback to guide the retrieval process. Our method is grounded on the clustering hypothesis, a well-known concept in information retrieval, that states that documents that are semantically similar to a relevant document tend to answer the same query (Jardine and van Rijsbergen, 1971). Recently in the IR literature, there have been initial attempts to exploit this idea to improve retrieval using better assessments from a re-ranker (MacAvaney et al., 2022; Kulkarni et al., 2023; Rathee et al., 2024).

We propose a neighborhood aware retrieval procedure that judiciously explores related documents of the initially retrieved documents leveraging the interim feedback of LLM-based answering system based on semantic uncertainty of potential answers. In our approach, SUNAR (**S**emantic **U**ncertainty based **N**eighborhood **A**ware **R**etrieval) we first compute a neighborhood graph based on semantic similarities between documents. During the query processing phase, we judiciously explore the neighborhood graph of the retrieved documents to use in the LLM prompt as context. Central to our method is the hypothesis that better document rankings correlate with lower uncertainty in the LLM-generated answer candidates. We quantify this uncertainty and use it as feedback to promote or penalize documents during retrieval dynamically. By integrating LLM-based signals into the neighborhood aware retrieval process, we effectively bootstrap the retrieval of relevant context, enhancing the overall system performance.

We conduct extensive experiments on two open-domain complex QA datasets to evaluate our approach. Our results show **significant improvements** over existing retrieve-and-reason baselines, with an increase in performance of up to **20%** on WQA and up to **31.84%** on MQA. To contextualize our performance, we close the performance gap with an idealized upper bound from 43% to 19% on WQA [1]. Our experiments also show that SUNAR can be retrofitted to LLM-based reasoning approaches (Yao et al., 2023; Xu et al., 2024), to improve their performance further.

2 Related Work

2.1 Complex Question Answering

Complex Question Answering tasks require multi-step reasoning by leveraging information from multiple sources (Ho et al., 2020; Trivedi et al., 2022; Mavi et al., 2022). HotPotQA (Yang et al., 2018), one of the first datasets introduced for multi-hop complex QA, which however has been found to not necessitate multi-hop reasoning (Min et al., 2019; Trivedi et al., 2022). To tackle this, more challenging datasets that require connected/multi-hop reasoning like 2WikiMultiHopQA (Ho et al., 2020) and MusiqueQA (Trivedi et al., 2022) have been proposed. Supervised and unsupervised (Perez et al., 2020) approaches for complex QA often rely on fine-tuning multiple specialized models to iteratively refine the query representations and arrive at final answer (Perez et al., 2020; Mavi

---

[1] An idealized upper bound is a QA system which has perfectly relevant results as context

et al., 2022). However, with the recent advances in Large Language Models (LLMs) (Wei et al., 2022a; Brown et al., 2020; Wei et al., 2022b), they are being adopted as general-purpose answer engines for complex tasks, obviating the need for fine-tuning of multiple models. However, relying only on parametric knowledge of LLMs may lead to sub-par performance due to factual inconsistencies resulting from hallucination (Ji et al., 2023).

2.2 Retrieval Approaches

The typical retrieval approaches for QA systems follow a retrieve-and-read framework (Chen et al., 2017), where the retriever can be either sparse (Robertson and Zaragoza, 2009; Formal et al., 2021) or dense (Karpukhin et al., 2020; Xiong et al., 2020; Song et al., 2020). The reader or the answering system can be either fine-tuned (Mavi et al., 2022) or employ few-shot or zero-shot reasoning (Brown et al., 2020; Lewis et al., 2021). Pseudo Relevance Feedback (PRF) approaches aim to bridge query-document vocabulary/representation mismatch and improve retrieval effectiveness by recomputing query representations for re-retrieval/re-ranking using relevant documents from first-stage retrieval (Li et al., 2022). However, PRF methods are known to suffer from *drift* resulting in suboptimal performance due to retrieving irrelevant contexts. Our proposed method combines standard retrieve and rerank systems with documents neighborhood information. Most similar to our work, adaptive retrieval approaches have been proposed (MacAvaney et al., 2022; Kulkarni et al., 2023; Rathee et al., 2024) that use rankers to guide the retrieval process while keeping the reranking costs constant. We are different from earlier work in adapting LLM feedback for neighborhood retrieval. Also different from PRF techniques, we operate on the document similarity dimension instead of similarity to the query representation.

2.3 Pipelines for complex QA

With the recent advances in LLMs, off-the-shelf Retrieval Augmented Generation (RAG) approaches have been proposed which retrieve and incorporate external knowledge to mitigate factual inconsistencies and generate accurate answers (Khot et al., 2023; Xu et al., 2024; Schick et al., 2023). These RAG pipelines are coupled with query understanding and reasoning approaches that leverage the emergent capabilities of the LLM to decompose a complex question into sub-questions and solve them iteratively (Press et al., 2023; Xu et al., 2024; Khot et al., 2023; Yao et al., 2023; Trivedi et al., 2023; Dua et al., 2022). However, the existing RAG pipelines for complex QA employ off-the-shelf retrievers for retrieving relevant knowledge without comprehensive evaluation and are limited by the performance of the retriever (Asai et al., 2024; Li et al., 2024). An additional challenge is the context limit of LLMs employed as readers, making it necessary to capture the relevant documents within the limited budget. Additionally, the inherent noise in retrieved documents, could result in a wrong reasoning path and (Shi et al., 2024) and derive a wrong answer. Hence, in this work, we focus on improving the recall of top-ranked documents that fit into the context limit of the reader LLMs through document Neighborhood Aware Retrieval enhanced with semantic uncertainty based LLM feedback.

3 Methodology

The common approach to performing open-domain QA tasks typically follows a two-step process: retrieval of relevant contexts, followed by a reasoning or QA step using those retrieved contexts. This is exemplified by traditional RAG pipelines (Lewis et al., 2021). For simpler or direct questions, this method is often sufficient. However, complex questions tend to consist of multiple sub-questions, which requires more advanced query understanding steps, such as decomposing the main question into sub-questions prior to retrieval. In this context, given a complex question $q \in Q$ and its corresponding answer $a \in A$, the decomposition process can be formally defined as: $de(q, a) = \{(sq_1, R_1, sa_1), \ldots, (sq_k, R_k, sa_k)\}$ where $k = k_{(q,a)}$ represents the number of sub-questions, (sq_i, sa_i) is the pair of sub-question and corresponding answer, and R_i is a list of top-l ranked documents (with $l \leq 10$) supporting the sub-question sq_i. After decomposing the complex question into sub-questions, the retrieval step is followed by a ranking phase, typically performed by a fine-tuned model. This ranking is critical due to the limited input budget of large language models (LLMs), which restricts the amount of information they can process at once. The ranking helps prioritize the most relevant documents for the final reasoning step. Finally, in the reasoning phase, the LLM processes the ranked and retrieved contexts to generate the final answer. In this work,

Algorithm 1 The SUNAR Algorithm

Input: Initial retrieved list R, batch size b, re-ranking budget c, document graph G
Output: Re-Ranked pool R^+
  1: $R^+ \leftarrow \emptyset$ ▷ Re-Ranking results
  2: $C \leftarrow R$ ▷ Candidate pool
  3: $N \leftarrow \emptyset$ ▷ Neighbor pool
  4: **do**
  5:     $B \leftarrow$ SCORE(top b from C, subject to c)
  6:     $\{sa_1...sa_m\} \leftarrow \phi(\mathbb{P}_{LLM}(sq_1, B))$
  7:     $\{ac_1..ac_s\} \leftarrow \sigma(sa_1..sa_m)$ ▷ Clustering
  8:
  9:     $B \leftarrow$ RESCORE(B,$1/s$)   ▷ Rescore batch
  10:    $R^+ \leftarrow R^+ \cup B$ ▷ Add batch to results
  11:
  12:    // Discard Batches
  13:    $R \leftarrow R \setminus B$
  14:    $N \leftarrow N \setminus B$
  15:    $N \leftarrow N \cup ($NEIGHBOURS$(B, G) \setminus R^+)$
  16:
  17:    //Alternate R and N
  18:    $C \leftarrow \begin{cases} R & \text{if } C = F \\ N & \text{if } C = N \end{cases}$
  19: **while** $|R^+| < c$

we utilize the LLM as the core question-answering model, responsible for reasoning over the retrieved and ranked contexts and providing the final answer. An overview of our approach is shown in Figure 2. In our approach, we employ a Neighborhood Aware Retrieval (NAR) method to overcome the bounded-recall issue of first-stage retrieval in classic RAG pipelines. To further ensure we capture the most relevant documents in the limited top-l budget we employ uncertainty-based LLM feedback to rescore the documents.

3.1 Neighborhood Aware Retrieval

Current RAG pipelines usually employ an off-the-shelf first-stage retriever for fetching relevant knowledge. While query understanding /question decomposition can help fetch better documents, the performance of complex QA is still limited by the quality of the first-stage retriever. To overcome the recall limitations of first-stage retrieval, we adopt Neighborhood Aware Retrieval (NAR) which employs the *Clustering Hypothesis* (Jardine and van Rijsbergen, 1971) which suggests that documents in the vicinity of highly scored documents tend to answer the same queries/questions.

3.1.1 Neighborhood Graph Construction

NAR first constructs a document-neighborhood graph $G = (V, E)$, also called the neighborhood graph in an offline phase. Each document is a

node and the top-k nearest neighbors are the edges, documents which are in the close vicinity (hence, $|E| = k|V|$) according to semantic relatedness. The neighborhood graph is constructed using a dense or sparse retriever. In the case of a dense retriever, for example, ColbertV2, the neighborhood graph can be constructed by using the representation space generated by the document encoder and fetching k-nearest neighbor documents for each document in the corpus. The neighborhood graph is constructed using a dense retriever if sparse retrieval is used for first-stage retrieval and vice versa to capture complementary signals. Note that the neighborhood graph construction is a one-time activity for each document collection, and during inference, the indexed graph is directly used for lookup for efficiency.

3.1.2 The SUNAR Algorithm

NAR starts with an initial set of ranked documents R obtained from first-stage retrieval. NAR scores the documents in batches, each of size b, in each iteration till budget c is reached (stopping condition in Line 15 of Algorithm 1. NAR comprises a dynamically updated candidate documents pool C (initialized to R, line 2) and a dynamically adapted neighbor pool (N, initially empty, line 3). Firstly, NAR employs a cross-encoder based re-ranker to score top b documents from C which constitute a batch B (line 5). After this step, in LLM guided NAR (lines 6-9) we employ LLM-based feedback to re-score the documents in batch B. While we further elaborate on this in Section 3.2, vanilla NAR does not include this re-scoring step and adds B to R^+ (line 9). Then the documents of the batch B are removed from N and R as they are already ranked. Following this step, the neighbors of documents in B are looked up from the neighborhood graph G. These documents, barring those already ranked are added to the neighbor pool (line 13) prioritized according to the ranking score of the source document in B. NAR then explores a batch of documents from neighbor pool N instead of the next batch from R. This ensures the final ranked list includes documents not only from R but additional documents also that are not included in the initial retrieval R. This aids in overcoming the recall limitations of first-stage retrieval. NAR proceeds by alternating between R and N till budget c is reached and the final ranked list R^+ is returned. Note that NAR is invoked for each sub-question sq_i.

3.2 LLM feedback for Neighborhood Aware Retrieval

To improve the re-ranking of documents and promote the most relevant documents in the limited top-l budget (for instance, where l=10), we propose an LLM-based feedback mechanism to re-score the top-ranked documents from the cross-encoder for each batch of NAR. While the cross-encoder aids in re-rank the retrieved documents for each sub-question from a relevance perspective, the LLM feedback helps re-score documents from an uncertainty perspective.

More formally, given the sub-question and ranked list of documents B from the batch in the current iteration of NAR (sq_1, B), where $B = [d_1, d_2, ..., d_b]$, the ASU (Answer Semantic Uncertainty) re-scores the returns the batch B with updated scores (6-9 of Algorithm 1).

$$B = \text{LLM\_SCORE}(sq_1, B) \qquad (1)$$

It accomplishes this by first estimating the consistency of the answers based on the number of answer semantic sets. This is computed by first estimating multiple answers for a given sub-question and ranked document list

$$\{sa_1..sa_m\} = \phi(\mathbb{P}_{LLM}(sq_1, R), m) \qquad (2)$$

where ϕ is the response generator that generates multiple responses $(sa_1...sa_m)$ from the distribution $\mathbb{P}_{LLM}$. We further define σ as the estimator of answer semantic sets through semantic equivalence estimation for semantic clustering. A semantic set is defined as a set of sequences that share the same meaning following the work (Kuhn et al., 2023). σ takes as input a set of answers and outputs the answer semantic sets.

$$\{ac_1..ac_s\} = \sigma(\{(sa_1.., sa_m)\}) \qquad (3)$$

where s denotes the number of semantic sets

It accomplishes this by clustering equivalent answers to the same answer semantic set through bi-directional entailment computed between each pair of answers. We employ bi-directional entailment as it is a stronger criterion for semantic equivalence (Kuhn et al., 2023). $\sigma(sa_i, sa_j)$ is defined as:

$$M_{NLI}(sa_i \rightarrow sa_j) \wedge M_{NLI}(sa_j \rightarrow sa_i)$$

The entailment condition is evaluated for each pair of sequences. If both directions hold true,

sa_i and sa_j are assigned to the same semantic set. If the entailment fails in either direction, the sequences are placed in distinct semantic sets. We posit that more semantic sets indicate that the LLM is uncertain about the answer to the given question and the current batch of documents. Finally, we re-score the documents in the current batch by penalizing the scores from the cross-encoder as follows:

$\text{LLM\_SCORE}(sq_1, B) = [\frac{sc_1}{s}..\frac{sc_n}{s}]$ where sc_i refers to the cross-encoder score for the ith document for a given sub-question in batch B. The above process is repeated for all batches till the budget c for NAR is reached for a given sub-question. This is followed by the answer generation step using the LLM by leveraging top-l (where l=10) documents from the re-ranked list. The same is repeated for all sub-questions till the decomposition stops and the final answer is generated.

However, we observe that in this sequential reasoning process, the final answer is derived solely based on answers to previous sub-questions and only the evidence for the last sub-question. Hence, any errors in intermediate steps could result in cascading errors resulting in a wrong final answer. Inspired by post-hoc LLM correction strategies, we propose a simple yet significantly effective Meta Evidence Reasoner (MER) to tackle this issue. The MER component leverages the reasoning path obtained through sequential reasoning $[(sq_1, sa_1...(sq_n, sa_n)]$ and the top-l evidences from the set of ranked list of documents $R_1^+, R_2^+...$ across sub-questions and prompts the LLM with the original question to obtain the final answer. The prompt is as shown in Figure 4 in Appendix F.

4 Experimental Setup

We answer the following research questions:

RQ I. Can NAR improve the recall for better complex QA?

RQ II. Can LLM-guided NAR improve overall complex QA performance compared to existing state-of-the-art?

RQ III. How well does SUNAR work across different LLM substrates and with different reasoning approaches?

Datasets: We experiment on well-known QA datasets that require compositional reasoning, namely MuSiQue (MQA) (Trivedi et al., 2022) and 2WikiMultiHopQa (WQA) (Ho et al., 2020). It has been observed that multi-hop datasets like Hot-

Method	MQA	WQA
Methods (w/o query understanding)		
ZERO-SHOT-COT (Kojima et al., 2023)	8.62	30.42
FEW-SHOT-COT (Wei et al., 2023)	15.02	32.83
FEW-SHOT-COT +PRF (Li et al., 2022)	16.69	35.55
SUNAR$_R$ (ours)	21.32	40.96
Methods (w/ query understanding)		
Self-RAG (Asai et al., 2024)	17.80	35.25
ReAct (Yao et al., 2023)	21.41	43.25
DecomP (Khot et al., 2023)	21.01	44.08
IRCoT (Trivedi et al., 2023)	24.12	42.01
SearChain (Xu et al., 2024)	21.72	44.42
SELF-ASK +PRF (Li et al., 2022)	20.28	44.58
SELF-ASK (Press et al., 2023)	24.84	45.75
NAR (w/ query understanding) (ours)		
SUNAR$_R$	28.11	47.67
SUNAR	**32.75** †	**54.83**†
Golden Evidence		
(Ideal Upper Bound)		
FEW-SHOT-COT	44.28	65.55

Table 1: Results across datasets. The model used for SUNAR and other approaches is gpt-3.5-turbo unless otherwise specified.† indicates statistical significance over SELF-ASK at 0.1 level.

potQA (Yang et al., 2018) do not necessitate connected reasoning and can be answered using single hop (Min et al., 2019; Trivedi et al., 2022). However, MQA and WQA are designed in a manner that mandates compositional/multi-step reasoning. Following (Press et al., 2023), we employ the 2-hop questions from MQA leading to 1252 questions for evaluation. For WQA we evaluate on the sampled 1200 questions following SELF-ASK (Press et al., 2023). Combining the contexts with distractors from MQA and WQA following the setup of (Khot et al., 2023) results in a corpus of 569461 documents.

Evaluation Metrics: We evaluate the QA performance using the well-known cover-EM metric (Rosset et al., 2021; Press et al., 2023) which checks whether the ground truth is contained in the generated answer. We also evaluate retrieval performance for the original questions as shown in Tables 7,8 using Normalized Discounted Cumulative Gain (nDCG@k) and Recall@k (R@k).

SUNAR details: We employ SPLADE-v2 for the first stage retrieval in SUNAR as it is better than other off-the-shelf dense retrievers usually employed in RAG pipelines as seen from results in Table 8 in **Appendix C**. The neighborhood graph for SUNAR is formed using ColbertV2 which is a late-interaction model and is better at capturing relevance between documents which are usually longer than queries. We perform a KNN search

over the corpus for each document and retrieve the top-100 documents as neighbors to form the neighborhood graph. In the NAR setup, we consider top-100 documents from the first stage retrieval and only 10 neighbors for the documents under consideration in the current iteration for a fair comparison with baselines that re-rank top-1000 documents from first stage retrieval (Table 9 in Appendix D and PRF baselines in Table 1). We use *nreimers/mmarco-mMiniLMv2-L12-H384-v1* as scorer in NAR pipeline and for re-ranking enhanced baselines. In the end, the top-10 documents are used from the ranked list for LLM reasoning to answer the question in SUNAR and all baselines. To observe the impact of variation on the number of documents we experiment with {1,3,5,7,10} documents for WQA and report the performance in Figure 3 in Appendix E. We observe top-10 to be optimal from the results. While, SUNAR uses SELF-ASK style decomposition for query understanding, we also show it generalizes to other approaches like SearChain and ReAct (Table 2). We also evaluate variations of SUNAR where, SUNAR$_R$ refers to w/o ASU and MER components.

LLM Details: We employ gpt-3.5-turbo as the backbone for the baselines and our approach for results shown in Table 1. We also evaluate the best baseline and our approach using alternate models like gpt-4o-mini and open-source models like LLama 3.1 and Mistral v0.2. We set max_tokens for output generation to 1000 and $frequency\_penalty$ to 0.8 and $presence\_penalty$ to 0.6 to reduce repetition. **Baselines:** We compare SUNAR with state-of-the-art pipelines that employ retrieval without query understanding (w/o indicates without) and approaches that employ question understanding. FEW-SHOT-COT (Wei et al., 2023) employs stepwise reasoning given the original question and evidence retrieved using SPLADEv2 for the original question. We also compare with state-of-the-art query understanding or reasoning approaches like SUNAR (Press et al., 2023), DecomP (Khot et al., 2023), SearChain (Xu et al., 2024) that decompose the original question and interactively query the retriever to obtain relevant contexts. We use SPLADEv2 for first-stage retrieval across all baselines for a fair comparison. While all baselines use retrieve-reason paradigm, we further enhance these baselines by re-ranking the contexts and results are reported in Appendix D (Table 9). We further elab-

orate on PRF baseline in Appendix B. All prompts are presented in Appendix F.

5 Results

5.1 Impact of Neighborhood aware retrieval

To answer **RQ1**, we evaluate the impact of NAR (neighborhood Aware Retrieval) to state-of-the-art approaches adopted for complex reasoning, as shown in Table 1. We observe that SUNAR_R which employs NAR outperforms all existing approaches, highlighting the ability of NAR to surface relevant documents in the top-l results where $l <= 10$. We observe that SUNAR_R significantly outperforms FEW-SHOT-COT+PRF in both w query understanding and w/o query understanding setups. This is primarily because in NAR document-document similarities are also captured, rather than depending only on query-document similarity. Hence, a document that is less related to the current query might be closer to another relevant document not currently captured in the top-l results. NAR helps capture such documents and re-score them to surface them in the top-l results. We posit that FEW-SHOT-COT+PRF underperforms due to the possibility of drift where the augmented query may capture only relevant documents for a particular aspect missing out on other relevant documents due to the multi-aspect nature of complex QA. We also observe that SUNAR_R outperforms other state-of-the-art approaches that perform decomposition. These approaches like SELF-ASK, ReAct and Searchain decompose the query into sub-questions that capture the multi-aspect nature of the query and perform better than approaches without decomposition. However, they still rely on query-document relevance estimates from first-stage retrieval and do not optimize for recall at top-l and hence fall short when compared to SUNAR_R which captures more relevant documents with the limited budget. Even when these query understanding baselines are enhanced with re-ranking, they fall short compared to SUNAR as observed from Table 9 due to the bounded-recall problem.

5.2 Significance of LLM guided NAR for complex QA

To address **RQ2**, we evaluate SUNAR to observe the impact of LLM guidance on NAR and downstream LLM reasoning for complex QA. As observed from Table 1, SUNAR significantly outperforms state-of-the-art approaches for complex QA

Method	MQA	WQA
w/o SUNAR		
ReAct	21.41	43.25
SearChain	21.72	44.42
w/ SUNAR		
ReAct	28.67	48.40
SearChain	**30.43**	**55.50**

Table 2: Performance comparison when augmenting SUNAR to other approaches (gpt-3.5-turbo).

Method	MQA	WQA
gpt-4o-mini		
SELF-ASK	26.76	37.33
SUNAR	**32.19**	**48.16**
Llama 3.1 (8B)		
SELF-ASK	5.43	25.83
SUNAR	**13.82**	**39.52**
Mistral v0.2 (7B)		
SELF-ASK	7.84	27.72
SUNAR	**26.12**	**40.23**

Table 3: Results across different LLM substrates.

Method	MQA	WQA
SUNAR	**32.75**	**54.83**
SUNAR (-ASU)	30.43	50.75
SUNAR_R (-ASU) (-MER)	28.11	47.67

Table 4: Ablations of SUNAR (gpt-3.5-turbo).

Retriever	MQA		WQA	
	R@1	R@10	R@1	R@10
SELF-ASK(Re-Rank)	0.157	0.309	0.230	0.402
SELF-ASK(Retrieval)	0.152	0.240	0.205	0.399
SUNAR	**0.284**	**0.459**	**0.287**	**0.606**

Table 5: Retrieval performances on MQA and WQA with query understanding.

and also outperforms SUNAR_R by about **15.02%** in WQA and by **16.9%** in MQA. We observe that this is primarily due to the LLM guidance in the form of Answer Semantic Uncertainty (ASU) based re-scoring of documents. This is also evident from retrieval performance reported in Table 5, where SUNAR outperforms SELF-ASK by **48.54 %** on MQA and by **50.75%** on WQA which also positively impacts downstream reasoning performance. While NAR (SUNAR_R) does offer improvements over traditional RAG pipelines for complex QA, it still relies on the notion of relevance from the cross-encoder employed as a re-ranker. SUNAR employs LLM based ASU measure as feedback that goes beyond semantic relatedness and penalizes documents leading to inconsistent answers (more

Method	Evidences
Question	**Where was the director of film Ronnie Rocket born?** [Dataset: **WQA**]
SELF-ASK	[Evidence 1]: This is a list of film series by director.
	[Evidence 2]: This is a list of notable directors in motion picture and television arts.
	[Final Answer]: Unknown
SUNAR (ours)	[Evidence 1]: Ronnie Rocket is an unfinished film project written by David Lynch, who also intended [. . .].
	[Evidence 2]: David Keith Lynch was born in Missoula, Montana, on January 20, 1946. His father [. . .] .
	[Final Answer]: Missoula, Montana
Question	**Who did the screenwriter for Good Will Hunting play in Dazed and Confused?** [Dataset: **MQA**]
SELF-ASK	[Evidence 1]: Damon begins working alongside his younger brother, Stefan Salvatore, to resist greater[. . .].
	[Evidence 2]: Damon Salvatore is a fictional character in The Vampire Diaries. He is portrayed by Ian Somerhalder in the television.
	[Final Answer]: Damon Salvatore
SUNAR (ours)	[Evidence 1]: Damon and Ben Affleck wrote G̈ood Will Hunting(̈1997), a screenplay[. . .].
	[Evidence 2]: Benjamin Affleck- Boldt(born August 15, 1972) is an American actor . He later appeared in the independent coming- of- age comedyD̈azed and Confused as Fred O'Bannion [. . .]"
	[Final Answer]: Fred O'Bannion

Table 6: Qualitative analysis of evidences retrieved by SELF-ASK and SUNAR

semantic sets). This helps reduce the number of distractors in top-l results and ASU also serves as a proxy of whether the evidences help provide consistent answers for the question. Other query understanding approaches like SearChain, ReAct, and SELF-ASK decompose the question and iteratively interact with the retriever to obtain relevant documents but do not handle the presence of distractors as they purely rely on semantic relatedness to the query. For instance, in our **qualitative analysis** in Table 6, we observe that on MQA SELF-ASK recovers documents irrelevant to answer the question but which have high semantic relatedness to the sub-question related to "Matt Damon". This results in wrong answer as they are distractors and do not answer the complex question. However, SUNAR is able to surface two of the most relevant documents that directly answer the question in the top-5 results. Hence, SUNAR outperforms other approaches owing to LLM guided feedback for retrieval through ASU based penalty and LLM guided feedback for reasoning through MER based post-hoc correction.

5.3 SUNAR with different LLM substrates and query understanding approaches

To answer **RQ3**, we compare SUNAR with the best baseline SELF-ASK on different LLM substrates such as gpt-4o-mini, Llama 3.1 (8B), and Mistral v0.2 (7B) to observe the generalization capabilities of SUNAR. From the results in Table 3, we observe that SUNAR performs significantly better, demonstrating that SUNAR is LLM agnostic and offers significant gains over existing state-of-the-art query understanding and RAG pipelines on smaller

open-source models. We observe that this is primarily due to the relevant evidence surfaced by LLM guided NAR with fewer distractors.

To demonstrate that SUNAR is agnostic to query understanding/reasoning approaches, we combine SUNAR with approaches like ReAct and SearChain. The results are as shown in Table 2. We observe that ReAct and SearChain augmented with SUNAR offer significant gains over their counterparts that use existing retrieval approaches.

5.4 Ablations of SUNAR

We perform several ablations to assess the impact of LLM feedback for retrieval (ASU) and LLM feedback for reasoning (MER). The results are as shown in Table 4. We observe that both ASU based penalty (LLM feedback for retrieval) and MER based post-hoc correction (LLM feedback for reasoning) are essential for superior performance on MQA and WQA as SUNAR outperforms the variant without ASU and SUNAR_R. We observe that removing LLM feedback for retrieval results in more distractors and hence a drop in performance. Further, SUNAR_R which employs only NAR without any form of LLM feedback has a significant drop in performance due to the presence of distractors in ranked documents from NAR and cascading errors in the stepwise reasoning due to lack of post-hoc correction using MER.

6 Conclusion

In this paper, we introduce a novel approach SUNAR to improve retrieval and reasoning for complex QA. We propose Neighborhood Aware Re-

trieval (NAR) to improve recall in retrieved documents for downstream reasoning. We further augment NAR with LLM based feedback to reduce the impact of distractors and improve downstream reasoning. We observe significant performance improvements over state-of-the-art approaches on different datasets across different LLMs. Furthermore, when SUNAR is applied to other reasoning/query understanding approaches it leads to significant performance improvements demonstrating that SUNAR is agnostic of query understanding approaches and LLMs. In the future, we plan to explore further efficient methods for the ASU component.

7 Limitations

Although SUNAR offers significant performance improvements over existing off-the-shelf RAG pipelines, there are a few limitations that offer avenues for future work. Though NAR is based on the philosophy of adaptive retrieval which is more efficient than exhaustive re-ranking, a more principled selection of candidates for ranking in line 18 of Algorithm 1 with early stopping criterion could make it more efficient. We defer this for future work as it is beyond the scope of current work. For instance, the LLM based feedback now relies on computing semantic sets, which serve as a proxy for consistency in LLM answers for a given set of evidences. More principled conformal prediction based approaches could be adopted to calibrate LLM confidence for answers generated and use the measure for penalizing/promoting documents. We also defer this for future work.

8 Ethical Considerations and Risks

We primarily propose a novel uncertainty based neighborhood aware retrieval approach for complex QA tasks. We use only publicly available QA datasets from Wikipedia that do not contain private or harmful information. While we employ LLMs for QA systems in our experiments, we do not prompt them in a format that would elicit harmful or biased information. All our prompts are reported in Appendix F. Our tool is intended to enhance answering engines to aid users in their complex questions.

References

Akari Asai, Zeqiu Wu, Yizhong Wang, Avirup Sil, and Hannaneh Hajishirzi. 2024. Self-RAG: Learning to retrieve, generate, and critique through self-reflection. In *The Twelfth International Conference on Learning Representations*.

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. 2020. Language models are few-shot learners. *Advances in neural information processing systems*, 33:1877–1901.

Danqi Chen, Adam Fisch, Jason Weston, and Antoine Bordes. 2017. Reading Wikipedia to answer open-domain questions. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1870–1879, Vancouver, Canada. Association for Computational Linguistics.

Dheeru Dua, Shivanshu Gupta, Sameer Singh, and Matt Gardner. 2022. Successive prompting for decomposing complex questions. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 1251–1265, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Thibault Formal, Benjamin Piwowarski, and Stéphane Clinchant. 2021. Splade: Sparse lexical and expansion model for first stage ranking. In *Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '21, page 2288–2292, New York, NY, USA. Association for Computing Machinery.

Xanh Ho, Anh-Khoa Duong Nguyen, Saku Sugawara, and Akiko Aizawa. 2020. Constructing a multi-hop QA dataset for comprehensive evaluation of reasoning steps. In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 6609–6625, Barcelona, Spain (Online). International Committee on Computational Linguistics.

N. Jardine and Cornelis Joost van Rijsbergen. 1971. The use of hierarchic clustering in information retrieval. *Inf. Storage Retr.*, 7(5):217–240.

Ziwei Ji, Nayeon Lee, Rita Frieske, Tiezheng Yu, Dan Su, Yan Xu, Etsuko Ishii, Ye Jin Bang, Andrea Madotto, and Pascale Fung. 2023. Survey of hallucination in natural language generation. *ACM Computing Surveys*, 55(12):1–38.

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 *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 6769–6781, Online. Association for Computational Linguistics.

Omar Khattab, Arnav Singhvi, Paridhi Maheshwari, Zhiyuan Zhang, Keshav Santhanam, Sri Vardhamanan, Saiful Haq, Ashutosh Sharma, Thomas T. Joshi, Hanna Moazam, Heather Miller, Matei Zaharia, and Christopher Potts. 2023. Dspy: Compiling declarative language model calls into self-improving pipelines.

Tushar Khot, Harsh Trivedi, Matthew Finlayson, Yao Fu, Kyle Richardson, Peter Clark, and Ashish Sabharwal. 2023. Decomposed prompting: A modular approach for solving complex tasks. In *The Eleventh International Conference on Learning Representations*.

Takeshi Kojima, Shixiang Shane Gu, Machel Reid, Yutaka Matsuo, and Yusuke Iwasawa. 2023. Large language models are zero-shot reasoners.

Lorenz Kuhn, Yarin Gal, and Sebastian Farquhar. 2023. Semantic uncertainty: Linguistic invariances for uncertainty estimation in natural language generation. In *The Eleventh International Conference on Learning Representations*.

Hrishikesh Kulkarni, Sean MacAvaney, Nazli Goharian, and Ophir Frieder. 2023. Lexically-accelerated dense retrieval. In *Proceedings of the 46th International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR 2023, Taipei, Taiwan, July 23-27, 2023*, pages 152–162. ACM.

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.

Hang Li, Shengyao Zhuang, Ahmed Mourad, Xueguang Ma, Jimmy Lin, and Guido Zuccon. 2022. Improving query representations for dense retrieval with pseudo relevance feedback: A reproducibility study. In *Advances in Information Retrieval: 44th European Conference on IR Research, ECIR 2022, Stavanger, Norway, April 10–14, 2022, Proceedings, Part I*, page 599–612, Berlin, Heidelberg. Springer-Verlag.

Junlong Li, Jinyuan Wang, Zhuosheng Zhang, and Hai Zhao. 2024. Self-prompting large language models for zero-shot open-domain QA. In *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pages 296–310, Mexico City, Mexico. Association for Computational Linguistics.

Sean MacAvaney, Nicola Tonellotto, and Craig Macdonald. 2022. Adaptive re-ranking with a corpus graph. In *Proceedings of the 31st ACM International Conference on Information & Knowledge Management, Atlanta, GA, USA, October 17-21, 2022*, pages 1491–1500. ACM.

Vaibhav Mavi, Anubhav Jangra, and Adam Jatowt. 2022. A survey on multi-hop question answering and generation.

Sewon Min, Eric Wallace, Sameer Singh, Matt Gardner, Hannaneh Hajishirzi, and Luke Zettlemoyer. 2019. Compositional questions do not necessitate multi-hop reasoning. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 4249–4257, Florence, Italy. Association for Computational Linguistics.

Rodrigo Nogueira, Wei Yang, Kyunghyun Cho, and Jimmy Lin. 2019. Multi-stage document ranking with bert.

Ethan Perez, Patrick Lewis, Wen-tau Yih, Kyunghyun Cho, and Douwe Kiela. 2020. Unsupervised question decomposition for question answering. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 8864–8880, Online. Association for Computational Linguistics.

Ofir Press, Muru Zhang, Sewon Min, Ludwig Schmidt, Noah A. Smith, and Mike Lewis. 2023. Measuring and narrowing the compositionality gap in language models.

Mandeep Rathee, Sean MacAvaney, and Avishek Anand. 2024. Quam: Adaptive retrieval through query affinity modelling. *arXiv preprint arXiv:2410.20286*.

Stephen Robertson and Hugo Zaragoza. 2009. The probabilistic relevance framework: Bm25 and beyond. *Foundations and Trends® in Information Retrieval*, 3(4):333–389.

Corby Rosset, Chenyan Xiong, Minh Phan, Xia Song, Paul Bennett, and Saurabh Tiwary. 2021. Knowledge-aware language model pretraining.

Timo Schick, Jane Dwivedi-Yu, Roberto Dessì, Roberta Raileanu, Maria Lomeli, Luke Zettlemoyer, Nicola Cancedda, and Thomas Scialom. 2023. Toolformer: Language models can teach themselves to use tools.

Zhengliang Shi, Shuo Zhang, Weiwei Sun, Shen Gao, Pengjie Ren, Zhumin Chen, and Zhaochun Ren. 2024. Generate-then-ground in retrieval-augmented generation for multi-hop question answering. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 7339–7353, Bangkok, Thailand. Association for Computational Linguistics.

Kaitao Song, Xu Tan, Tao Qin, Jianfeng Lu, and Tie-Yan Liu. 2020. Mpnet: masked and permuted pre-training for language understanding. In *Proceedings of the 34th International Conference on Neural Information Processing Systems*, NIPS '20, Red Hook, NY, USA. Curran Associates Inc.

Harsh Trivedi, Niranjan Balasubramanian, Tushar Khot, and Ashish Sabharwal. 2022. MuSiQue: Multi-hop questions via single-hop question composition.

Transactions of the Association for Computational Linguistics, 10:539–554.

Harsh Trivedi, Niranjan Balasubramanian, Tushar Khot, and Ashish Sabharwal. 2023. Interleaving retrieval with chain-of-thought reasoning for knowledge-intensive multi-step questions. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 10014–10037, Toronto, Canada. Association for Computational Linguistics.

Jason Wei, Yi Tay, Rishi Bommasani, Colin Raffel, Barret Zoph, Sebastian Borgeaud, Dani Yogatama, Maarten Bosma, Denny Zhou, Donald Metzler, Ed H. Chi, Tatsunori Hashimoto, Oriol Vinyals, Percy Liang, Jeff Dean, and William Fedus. 2022a. Emergent abilities of large language models.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Brian Ichter, Fei Xia, Ed Chi, Quoc Le, and Denny Zhou. 2023. Chain-of-thought prompting elicits reasoning in large language models.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Brian Ichter, Fei Xia, Ed H. Chi, Quoc V. Le, and Denny Zhou. 2022b. Chain-of-thought prompting elicits reasoning in large language models. In *Advances in Neural Information Processing Systems 35: Annual Conference on Neural Information Processing Systems 2022, NeurIPS 2022, New Orleans, LA, USA, November 28 - December 9, 2022*.

Lee Xiong, Chenyan Xiong, Ye Li, Kwok-Fung Tang, Jialin Liu, Paul Bennett, Junaid Ahmed, and Arnold Overwijk. 2020. Approximate nearest neighbor negative contrastive learning for dense text retrieval.

Shicheng Xu, Liang Pang, Huawei Shen, Xueqi Cheng, and Tat-Seng Chua. 2024. Search-in-the-chain: Interactively enhancing large language models with search for knowledge-intensive tasks.

Zhilin Yang, Peng Qi, Saizheng Zhang, Yoshua Bengio, William 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*, pages 2369–2380, Brussels, Belgium. Association for Computational Linguistics.

Shunyu Yao, Jeffrey Zhao, Dian Yu, Nan Du, Izhak Shafran, Karthik R Narasimhan, and Yuan Cao. 2023. React: Synergizing reasoning and acting in language models. In *The Eleventh International Conference on Learning Representations*.

A Dataset

We evaluate on MusiqueQA and 2WikimultihopQA released under CC-BY-4.0 LICENSE

2WikiMultihopQA (WQA): Wikipedia based multi-hop questions that consists of compositional queries and comparison based queries. We evaluate on 1200 compositional questions as done in self-ask (Press et al., 2023) for a fair comparison. Example Question: Who was born later, Gideon Johnson or Holm Jølsen?

MuSiQue (MQA) Comprises hard multi-hop compositional questions constructed from Wikipedia-based QA datasets. The dataset was constructed from multiple single-hop datasets to result in complex questions that mandate multistep reasoning. MQA is a challenging dataset created to avoid possible shortcuts in existing multi-hop datasets like HotPotQA, where models do not perform compositional reasoning. We evaluate on 1252 compositional questions as done in self-ask (Press et al., 2023) for a fair comparison. Example Question: What did the actress in My Fair Lady win a Tony for ?

B Baselines

PRF (w/o query understanding): We follow ANCE-PRF (Li et al., 2022), where query representations are updated using all documents from first-stage retrieval. This makes a faulty assumption that top documents in first-stage retrieval are relevant. We make this much stronger by identifying golden documents directly from SPALDEv2 results and concatenate them to query. This updated query is encoded using ColBERTv2 to update query representations and employed to re-rank results from first-stage results. Note that this is an ideal scenario where gold documents are looked up to formulate a strong baseline to compare to SUNAR.

PRF (w query understanding): It is similar to the above setup further augmented with query understanding where the retrieval happens for each sub-question obtained from original question and the query representation is recomputed using gold documents from this combined retrieval pool across sub-questions. Using the new query representation, the combined retrieval pool is re-ranked using *nreimers/mmarco-mMiniLMv2-L12-H384-v1* for fair comparison with SUNAR.

Retriever	MQA		WQA	
	nDCG	R@10	nDCG	R@10
SPLADEV2	0.155	0.062	0.251	0.186
PRF	0.183	0.065	0.271	0.190
SUNAR$_R$	0.179	**0.073**	**0.297**	**0.216**

Table 7: Retrieval performances on MQA and WQA w/o query understanding.

Retriever	MQA		WQA	
	nDCG	R@100	nDCG@10	R@100
SPLADeV2	**0.155**	**0.181**	**0.251**	**0.323**
DPR	0.109	0.107	0.126	0.179
ANCE	0.140	0.115	0.212	0.223
Tas-b	0.176	0.172	0.277	0.303
MPNet	0.163	0.149	0.222	0.253
Contriever	0.155	0.169	0.216	0.294

Table 8: Performance of different retrieval models on MQA and WQA when documents are retrieved using original question.

C Comparison of Retrievers

We evaluate different retrieval approaches for the original questions in WQA and MQA and report nDCG@10 and Recall@100 as shown in Table 8. We also compare SUNAR$_R$ with PRF in w/o query understanding setup as observed in Table 7. We observe that surprisingly sparse retrievers like SPLADEv2 provide superior performance as measured by nDCg@10 and Recall@100. This indicates that SPLADE is able to capture more relevant documents as measured by recall, and is also able to rank them higher as measured by nDCG compared to other retrievers. Hence, we employ SPLADEv2 as first-stage retriever for our approach and all baselines in this work for a fair comparison. From Table 7 we also observe that vanilla version of SUNAR w/o query understanding still outperforms PRF based approaches which are limited due to possibility of query drift.

D Comparison with baselines with Re-Ranking

The existing state-of-the-art query understanding approaches decompose a complex question and interactively query a retriever to get relevant evidence, followed by LLM based reasoning. We further enhance these baselines by re-ranking the contexts/evidences retrieved for each sub-question. We use *nreimers/mmarco-mMiniLMv2-L12-H384-*

Method	MQA	WQA
Methods (w/ query understanding)		
ReAct (Yao et al., 2023)	19.65	42.75
DecomP (Khot et al., 2023)	22.66	44.62
SearChain (Xu et al., 2024)	25.71	45.16
SELF-ASK (Press et al., 2023)	24.84	45.83
NAR (w/ query understanding) (ours)		
SUNAR_R	28.11	47.67
SUNAR	**32.75** †	**54.83**†

Table 9: Results across datasets where baselines employ re-ranking for each sub-question after first-stage retrieval. The model used for SUNAR and other approaches is gpt-3.5-turbo unless otherwise specified.† indicates statistical significance over SELF-ASK at 0.1 level.

v1 as the ranker same as the scorer used in NAR for SUNAR to ensure a fair comparison. We observe, from comparing the results in Table 9 and Table 1, that re-ranking provided slight performance gains but not significant enough due to the bounded-recall problem where re-ranking is still limited by recall of documents retrieved in first stage using SPLADEv2. This further strengthens our hypothesis that NAR (SUNAR_R and LLM guided NAR (SUNAR) help bridge the recall gap and lead to significant performance improvements for complex QA.

E   Variation of number of documents

We vary the number of top-l documents used for downstream reasoning with LLM for SUNAR, where $l = \{1, 3, 5, 7, 10\}$ for WQA as shown in Figure 3. We observe that $l = 10$ yields the best performance. We also observe that performance variation across different values of l is not huge for SUNAR demonstrating that SUNAR captures relevant documents in top-10 results compared to other approaches. We also use top-10 documents for LLM reasoning for all other baselines for a fair comparison.

F   Prompts

We present the prompts used for the baselines and SUNAR for the query understanding, which aids in reasoning. These prompts are used for all LLMs. The prompt employed for SELF-ASK along with demonstration samples/exemplars are as shown in Figure 6 (WQA) and Figure 5 (MQA), meta-reasoner prompt template (Figure 4), SearChain (Figure 7) and DecomP (Figure 8 and ReAct 9. We use a prompt similar to SELF-ASK
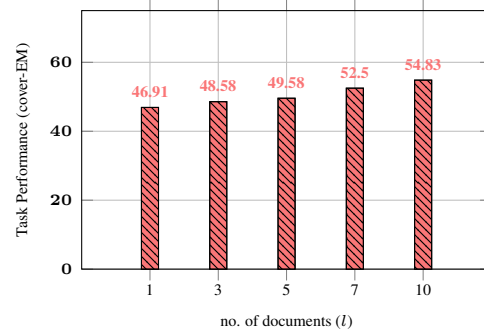

Figure 3: QA performance for different values of l in top-l documents used for reasoning where $l=\{1,3,5,7,10\}$. $l=10$ works best.

Meta-Reasoner Prompt

Instruction:Follow the given examples and Given the question and context, reasoning path, think step by step extract key segments from given evidence relevant to question and give rationale, by forming your own reasoning path preceded by [Answer]: and output final answer for the question using information from given evidences and give concise precise answer preceded by [Final Answer]:.

Exemplars : {}

Given the above examples and
Existing reasoner path: {Reasoning path from SUNAR_R},
the evidence : {top-l evidences across sub-questions}
and use the most relevant information for the question from the most relevant evidence from given Evidence: and form your own correct reasoning path to derive the answer thinking step by step preceded by [Answer]: and subsequently give final answer as shown in above examples preceded by [Final Answer]: for
the Question: {Test Question}

Figure 4: Meta-reasoner prompt

for SUNAR main results in Table 1, but we also demonstrate the generality of our approach by applying query understanding approaches ReAct and SearChain in Table 2.

SELF-ASK MusiqueQA Prompt

Instruction:`Follow the given examples and Given the question determine if followup questions are needed and decompose the original question.`

Exemplars :

Question: When does monsoon season end in the state the area code 575 is located?
Are follow up questions needed here: Yes.
Follow up: Which state is the area code 575 located in?
Intermediate Answer: The area code 575 is located in New Mexico.
Follow up: When does monsoon season end in New Mexico?
Intermediate Answer: Monsoon season in New Mexico typically ends in mid-September.
[Final Answer]: mid-September.

Question: What is the current official currency in the country where Ineabelle Diaz is a citizen?
Are follow up questions needed here: Yes.
Follow up: Which country is Ineabelle Diaz a citizen of?
Intermediate Answer: Ineabelle Diaz is from Peurto Rico, which is in the United States of America.
Follow up: What is the current official currency in the United States of America?
Intermediate Answer: The current official currency in the United States is the United States dollar. [Final Answer]: United States dollar.

Question: Where was the person who founded the American Institute of Public Opinion in 1935 born? Are follow up questions needed here: Yes. Follow up: Who founded the American Institute of Public Opinion in 1935? Intermediate Answer: George Gallup. Follow up: Where was George Gallup born? Intermediate Answer: George Gallup was born in Jefferson, Iowa.
[Final Answer]: Jefferson.

Question: What is the sports team the person played for who scored the first touchdown in Superbowl 1?
Are follow up questions needed here: Yes.
Follow up: Which player scored the first touchdown in Superbowl 1?
Intermediate Answer: Max McGee.
Follow up: Which sports team did Max McGee play for?
Intermediate Answer: Max McGee played for the Green Bay Packers.
[Final Answer]: Green Bay Packers.

Question: The birth country of Jayantha Ketagoda left the British Empire when?
Are follow up questions needed here: Yes.
Follow up: What is the birth country of Jayantha Ketagoda?
Intermediate Answer: Sri Lanka.
Follow up: When did Sri Lanka leave the British Empire?
Intermediate Answer: Sri Lanka left the British Empire on February 4, 1948.
[Final Answer]: February 4, 1948

Input : Based on above examples, given the Question: determine, Are followup questions needed here ?

Figure 5: Example of In-context learning for MusiqueQA (MQA) through SELF-ASK based prompting of LLMs

SELF-ASK 2WikiMultiHopQA Prompt

Instruction:`Follow the given examples and Given the question determine if followup`
`questions are needed and decompose the original question.`

Exemplars :
Question: Who lived longer, Theodor Haecker or Harry Vaughan Watkins?
Are follow up questions needed here: Yes.
Follow up: How old was Theodor Haecker when he died?
Intermediate Answer: Theodor Haecker was 65 years old when he died.
Follow up: How old was Harry Vaughan Watkins when he died?
Intermediate Answer: Harry Vaughan Watkins was 69 years old when he died.
[Final Answer]: Harry Vaughan Watkins.
Question: Why did the founder of Versus die?
Are follow up questions needed here: Yes.
Follow up: Who founded Versus?
Intermediate Answer: Gianni Versace.
Follow up: Why did Gianni Versace die?
Intermediate Answer: Gianni Versace was shot and killed on the steps of his Miami Beach mansion on July 15, 1997.
[Final Answer]: Shot.
Question: Who is the grandchild of Dambar Shah?
Are follow up questions needed here: Yes.
Follow up: Who is the child of Dambar Shah?
Intermediate Answer: Dambar Shah (? - 1645) was the king of the Gorkha Kingdom. He was the father of Krishna Shah.
Follow up: Who is the child of Krishna Shah?
Intermediate Answer: Krishna Shah (? - 1661) was the king of the Gorkha Kingdom. He was the father of Rudra Shah.
[Final Answer]: Rudra Shah.
Question: Are both director of film FAQ: Frequently Asked Questions and director of film The Big Money from the same country?
Are follow up questions needed here: Yes.
Follow up: Who directed the film FAQ: Frequently Asked Questions?
Intermediate Answer: Carlos Atanes.
Follow up: Who directed the film The Big Money? Intermediate Answer: John Paddy Carstairs. Follow up: What is the nationality of Carlos Atanes? Intermediate Answer: Carlos Atanes is Spanish. Follow up: What is the nationality of John Paddy Carstairs?
Intermediate Answer: John Paddy Carstairs is British. [Final Answer]: No.
Question: Who was the maternal grandfather of George Washington?
Are follow up questions needed here: Yes. Follow up: Who was the mother of George Washington?
Intermediate answer: The mother of George Washington was Mary Ball Washington.
Follow up: Who was the father of Mary Ball Washington?
Intermediate answer: The father of Mary Ball Washington was Joseph Ball. [Final Answer]: Joseph Ball

Input : Based on above examples, given the Question: determine, Are followup questions needed here ?

Figure 6: Example of In-context learning for 2WikiMultiHopQA through SELF-ASK based prompting of LLMs

---

SearChain Prompt

Instruction: Solve a question answering task with interleaving Thought, Action, Observation steps. Thought can reason about the current situation, and Action can be three types: (1) Search[entity], which searches the exact entity on Wikipedia and returns the first paragraph if it exists. If not, it will return some similar entities to search.
(2) Lookup[keyword], which returns the next sentence containing keyword in the current passage.
(3) Finish[answer], which returns the answer and finishes the task.

Exemplars :
Question: When does monsoon season end in the state the area code 575 is located?
Are follow up questions needed here: Yes.
Follow up: Which state is the area code 575 located in?
Intermediate Answer: The area code 575 is located in New Mexico.
Follow up: When does monsoon season end in New Mexico?
Intermediate Answer: Monsoon season in New Mexico typically ends in mid-September.
[Final Answer]: mid-September.

Construct a global reasoning chain for this complex [Question] : " {} " You should generate a query to the search engine based on what you already know at each step of the reasoning chain, starting with [Query].
If you know the answer for [Query], generate it starting with [Answer].
You can try to generate the final answer for the [Question] by referring to the [Query]-[Answer] pairs, starting with [Final Answer].
If you don't know the answer, generate a query to search engine based on what you already know and do not know, starting with [Unsolved Query].
For example: [Question]: "Where do greyhound buses that are in the birthplace of Spirit If...'s performer leave from? "
[Query 1]: Who is the performer of Spirit If... ?
If you don't know the answer:
[Unsolved Query]: Who is the performer of Spirit If... ?
If you know the answer: [Answer 1]: The performer of Spirit If... is Kevin Drew. [Query 2]: Where was Kevin Drew born?
If you don't know the answer: [Unsolved Query]: Where was Kevin Drew born?
If you know the answer: [Answer 2]: Toronto.
[Query 3]: Where do greyhound buses in Toronto leave from?
If you don't know the answer: [Unsolved Query]: Where do greyhound buses in Toronto leave from?
If you know the answer: [Answer 3]: Toronto Coach Terminal.
[Final Content]: The performer of Spirit If... is Kevin Drew [1]. Kevin Drew was born in Toronto [2]. Greyhound buses in Toronto leave from Toronto Coach Terminal [3]. So the final answer is Toronto Coach Terminal.
[Final Answer]: Toronto Coach Terminal
[Question]:"Which magazine was started first Arthur's Magazine or First for Women?"
[Query 1]: When was Arthur's Magazine started?
[Answer 1]: 1844.
[Query 2]: When was First for Women started? [Answer 2]: 1989
[Final Content]: Arthur's Magazine started in 1844 [1]. First for Women started in 1989 [2]. So Arthur's Magazine was started first. So the answer is Arthur's Magazi
[Final Answer]: Arthur's Magazi. [. . .]

Figure 7: Searchain prompt used for complex QA (Xu et al., 2024)

Instruction:`Solve a question answering task with interleaving retrieval Follow the given examples and Given the question and context, use relevant evidence for each subquestion from given list of evidences, answer the subquestion and output final answer for the question using information in the context, answers to subquestions and give final answer strictly preceded by [Final Answer]:`

Exemplars :

[Question]: In which country did this Australian who was detained in Guantanamo Bay detention camp and published Guantanamo: My Journey receive para-military training?
Subquestion 1: (select) [retrieve odqa] Who is the Australian who was detained in Guantanamo Bay detention camp and published "Guantanamo: My Journey"?
Answer: "titles": ["Guantanamo: My Journey", "Bismullah v. Gates", "Guantanamo Bay detention camp"], "answer": ["David Hicks"]
Subquestion 1: (select) [retrieve odqa] In which country did David Hicks receive his para-military training?
Answer: "titles": ["John Adams Project", "Camp Echo (Guantanamo Bay)", "Guantanamo Bay Museum of Art and History", "David Hicks"], "answer": ["Afghanistan"]
Subquestion 3: (select) [multihop titleqa] Titles: ["Guantanamo: My Journey", "Bismullah v. Gates", " Guantanamo Bay detention camp", "John Adams Project", "Camp Echo (Guantanamo Bay)", "Guantanamo Bay Museum of Art and History", "David Hicks"].
Question: In which country did this Australian who was detained in Guantanamo Bay detention camp and published " Guantanamo My Journey" receive para-military training?
[Final Answer]: Afghanistan

[Question]: who is older Jeremy Horn or Renato Sobral ? Subquestion 1: (select) [retrieve odqa] When was Jeremy Horn born?
Answer: "titles": ["Zaza Tkeshelashvili", "Jeremy Horn", "Jeremy Horn (singer)", "Ricardo Arona"], " answer": ["August 25, 1975"]
Subquestion 2: (select) [retrieve odqa] When was Renato Sobral born?
Answer: "titles": ["Brian Warren", "Renato Sobral"], "answer": ["September 7, 1975"]
Subquestion 3: (select) [multihop titleqa] Titles: ["Zaza Tkeshelashvili", "Jeremy Horn", "Jeremy Horn (singer)", "Ricardo Arona", "Brian Warren", "Renato Sobral"]. Question: who is older Jeremy Horn or Renato Sobral ?
[Final Answer]: Jeremy Horn

[Question]: What was the 2014 population of the city where Lake Wales Medical Center is located?
Subquestion 1: (select) [retrieve odqa] Lake Wales Medical Center is located in what city?
Answer: "titles": ["Baylor College of Medicine", "Lake Wales Medical Center", "Tufts University School of Medicine", "Hanford Community Medical Center"], "answer": ["Polk County, Florida"]
Subquestion 2: (select) [retrieve odqa] What was the population of Polk County in 2014?
Answer: "titles": ["Banner University Medical Center Tucson", "Lake Wales, Florida"], "answer": ["15,140"]
Subquestion 3: (select) [multihop titleqa] Titles: ["Baylor College of Medicine", "Lake Wales Medical Center ", "Tufts University School of Medicine", "Hanford Community Medical Center", "Banner University Medical Center Tucson", "Lake Wales, Florida"]. Question: What was the 2014 population of the city where Lake Wales Medical Center is located?
[Final Answer]: 15,140

[Question]: Nobody Loves You was written by John Lennon and released on what album that was issued by Apple Records, and was written, recorded, and released during his 18 month separation from Yoko Ono?
Subquestion 1: (select) [retrieve odqa] What album was issued by Apple Records, and written, recorded, and released during John Lennon's 18 month separation from Yoko Ono?
Answer: "titles": ["John Lennon/Plastic Ono Band", "Milk and Honey (album)", "Walls and Bridges"], "answer": ["Walls and Bridges"]
Subquestion 2: (select) [retrieve odqa] Nobody Loves You was written by John Lennon on what album?
Answer: "titles": ["John Lennon Museum", "Nobody Loves You (When You're Down and Out)"], " answer": ["Walls and Bridges"]
Subquestion 3: (select) [multihop titleqa] Titles: ["John Lennon/Plastic Ono Band", "Milk and Honey (album)
[Final Answer]: Walls and Bridges [. . .]

Input: "Follow the above examples, and use the given Evidence: "+{}+" and use the information from the evidence to answer the original question and give final answer strictly preceded by [Final Answer]: for the Question: {}

Figure 8: DeComP prompt (Khot et al., 2023)

ReAct Prompt

Instruction:`Solve a question answering task with interleaving Thought, Action,`
`Observation steps.  Thought can reason about the current situation, and`
`Action can be three types:  (1) Search[entity], which searches the exact`
`entity on Wikipedia and returns the first paragraph if it exists.  If not,`
`it will return some similar entities to search.  (2) Lookup[keyword], which`
`returns the next sentence containing keyword in the current passage.  (3)`
`Finish[answer], which returns the answer and finishes the task.:`

Exemplars   :
Question: What is the elevation range for the area that the eastern sector of the Colorado orogeny extends into?
Thought 1: I need to search Colorado orogeny, find the area that the eastern sector of the Colorado orogeny extends into, then find the elevation range of the area.
Action 1: Search[Colorado orogeny]
Observation 1: The Colorado orogeny was an episode of mountain building (an orogeny) in Colorado and surrounding areas.
Thought 2: It does not mention the eastern sector. So I need to look up eastern sector.
Action 2: Lookup[eastern sector]
Observation 2: (Result 1 / 1) The eastern sector extends into the High Plains and is called the Central Plains orogeny.
Thought 3: The eastern sector of Colorado orogeny extends into the High Plains. So I need to search High Plains and find its elevation range.
Action 3: Search[High Plains]
Observation 3: High Plains refers to one of two distinct land regions:
Thought 4: I need to instead search High Plains (United States).
Action 4: Search[High Plains (United States)]
Observation 4: The High Plains are a subregion of the Great Plains. From east to west, the High Plains rise in elevation from around 1,800 to 7,000 ft (550 to 2,130 m).[3]
Thought 5: High Plains rise in elevation from around 1,800 to 7,000 ft, so the answer is 1,800 to 7,000 ft.
Action 5: Finish[1,800 to 7,000 ft]
[Final Answer]: 1,800 to 7,000 ft
Question: Musician and satirist Allie Goertz wrote a song about the T̈he Simpsonsc̈haracter Milhouse, who Matt Groening named after who?
Thought 1: The question simplifies to T̈he Simpsonsc̈haracter Milhouse is named after who. I only need to search Milhouse and find who it is named after.
Action 1: Search[Milhouse]
Observation 1: Milhouse Mussolini Van Houten is a recurring character in the Fox animated television series The Simpsons voiced by Pamela Hayden and created by Matt Groening.
Thought 2: The paragraph does not tell who Milhouse is named after, maybe I can look up n̈amed after.̈
Action 2: Lookup[named after]
Observation 2: (Result 1 / 1) Milhouse was named after U.S. president Richard Nixon, whose middle name was Milhous.
Thought 3: Milhouse was named after U.S. president Richard Nixon, so the answer is Richard Nixon.
Action 3: Finish[Richard Nixon]
Question: Which documentary is about Finnish rock groups, Adam Clayton Powell or The Saimaa Gesture?
Thought 1: I need to search Adam Clayton Powell and The Saimaa Gesture, and find which documentary is about Finnish rock groups.
Action 1: Search[Adam Clayton Powell]
Observation 1: Could not find [Adam Clayton Powell]. Similar: ['Adam Clayton Powell III', 'Seventh Avenue (Manhattan)', 'Adam Clayton Powell Jr. State Office Building', 'Isabel Washington Powell', 'Adam Powell', 'Adam Clayton Powell (film)', 'Giancarlo Esposito'].
Thought 2: To find the documentary, I can search Adam Clayton Powell (film).
Action 2: Search[Adam Clayton Powell (film)]
Observation 2: Adam Clayton Powell is a 1989 American documentary film directed by Richard Kilberg.
The film is about the rise and fall of influential African-American politician Adam Clayton Powell Jr.[3][4] It was later aired as part of the PBS series The American Experience.
Thought 3: Adam Clayton Powell (film) is a documentary about an African-American politician, not Finnish rock groups. So the documentary about Finnish rock groups must instead be The Saimaa Gesture.
Action 3: Finish[The Saimaa Gesture]
[Final Answer]: The Saimaa Gesture

[. . .]

Input: "Follow the above examples, and use the given Evidence: "+{}+"
and use the information from the evidence to answer the original question and give final answer strictly preceded by
[Final Answer]: for the Question: {}

Figure 9: ReAct prompt (Khot et al., 2023)