FalseCoTQA: Adversarial Multi-Hop QA via Knowledge-Grounded False Chains of Thought

Abstract

Multi-hop question answering (QA) models excel at decomposing complex queries into sequential reasoning steps, yet they remain vulnerable to subtly flawed inference chains that appear reasonable but are factually incorrect. To quantify and address this weakness, we present FalseCoTQA, an adversarial benchmark that injects knowledge-grounded false reasoning into retrieval-augmented contexts. Unlike prior methods that merely tweak surface text, FalseCoTQA leverages a domain-agnostic knowledge graph to systematically replace entities to construct semantically coherent yet incorrect chains of thought on top of standard multi-hop datasets (HotpotQA, and MuSiQue). By evaluating state-of-the-art language models on this benchmark, we observe dramatic drops in answer accuracy, highlighting their tendency to follow deceptive reasoning without verifying factual consistency. We expect the proposed benchmark to contribute to the evaluation and improvement of the robustness and reliability of language models in multi-hop question answering.

Keywords

Multi-hop Question Answering, Adversarial Benchmark, Chains of Thought, Knowledge-Grounded Fabrication, Retrieval-Augmented

ACM Reference Format:

. 2018. FalseCoTQA: Adversarial Multi-Hop QA via Knowledge-Grounded False Chains of Thought. In *Proceedings of 3rd International ACM SIGIR Conference on Information Retrieval in the Asia Pacific (SIGIR-AP '25).* ACM, New York, NY, USA, 8 pages. https://doi.org/XXXXXXX.XXXXXXX

1 Introduction

Multi-hop question answering (MHQA) tasks require models to integrate information from multiple, disparate evidence sources to arrive at a correct answer. While large language models (LLMs) can perform multi-step inference using their parametric weights, they suffer from static knowledge limitations and a propensity to hallucinate [22]. Retrieval-augmented language models (RALMs) address these shortcomings by fetching up-to-date evidence from external corpora, diverse data sources and descriptive metadata, thereby grounding outputs in factual information and reducing overconfident or incorrect generations [5, 9, 13, 20, 23]. Despite these benefits, RALMs tend to accept retrieved passages uncritically, treating them as inherently trustworthy and often bypassing any verification step [2, 7, 37]. In MHQA, where correct answers hinge on synthesizing multiple sources, this blind trust can propagate errors: an unreliable chain-of-thought (CoT) initiation leads directly to flawed reasoning [25], and noisy or adversarial retrievals can derail inference [4, 18]. Recent efforts have refined CoT prompting to more tightly bind reasoning steps to retrieved context, improving logical consistency in MHQA [19]. Parallel lines of work have measured and enhanced LLM robustness against irrelevant or misleading contexts [27, 30, 32, 34], where specifically in information retrieval systems, robustness to adversarial still serves as a remaining challenge [14]. Moreover, adversarial benchmarks have been proposed that alter retrieved facts to test model reliability under deceptive inputs [21, 35], some even extend this idea to altering the questions themselves [16]. However, these benchmarks often rely on LLM-generated samples lacking real-world grounding or coherent reasoning structure [1, 3, 11].

Crucially, prior attacks have focused on corrupting context passages, yet the potential to mislead models via fabricated CoTs remains underexplored [26]. With CoT prompting now central to RALM performance, we ask: *Can we craft semantically coherent but factually false reasoning trajectories that distract models more effectively than noisy documents?* We hypothesize that such *false* CoTs pose a more potent threat, enticing models into a believable inference path that ultimately yields wrong answers [27, 28, 36]. To investigate this vulnerability, we introduce FalseCoTQA, a new adversarial benchmark and framework powered by our Latent Twin Retrieval (LTR) algorithm. We apply LTR to HotpotQA [31], and MuSiQue [6], retrieving *latent twins* which are knowledge-grounded, semantically related entities, to replace originals in reasoning chains [12]. Unlike prior entity swaps based on surface similarity [1], LTR ensures replacements reflect real-world knowledge, making fabricated CoTs highly viable. This builds upon previous work on leveraging knowledge sources for entity-centric textual relations and reasoning [8, 15]. We then use an LLM to generate false CoTs incorporating these swaps and inject them into RALM prompts. Empirically, such *fabricated* CoTs produce significantly larger accuracy drops than noise-only context manipulations. As a complementary baseline, we also craft adversarial context paragraphs via LTR.

Through extensive evaluations on multiple LLMs, we demonstrate that FalseCoTQA, particularly with semantically grounded false CoTs, induces significantly larger degradation in answer accuracy than previous adversarial benchmarks relying on factually incorrect or noisy contexts [1]. This heightened vulnerability underscores the need for RALMs that are robust to adversarial reasoning in both CoT reasoning, and retrieved contexts. The observed performance drop highlights specific failure modes, helping guide the development of models resilient to reasoning errors and contextual noise.

SIGIR-AP '25, Xi'an, China

ACM ISBN 978-1-4503-XXXX-X/2018/06
https://doi.org/XXXXXXX.XXXXXXX

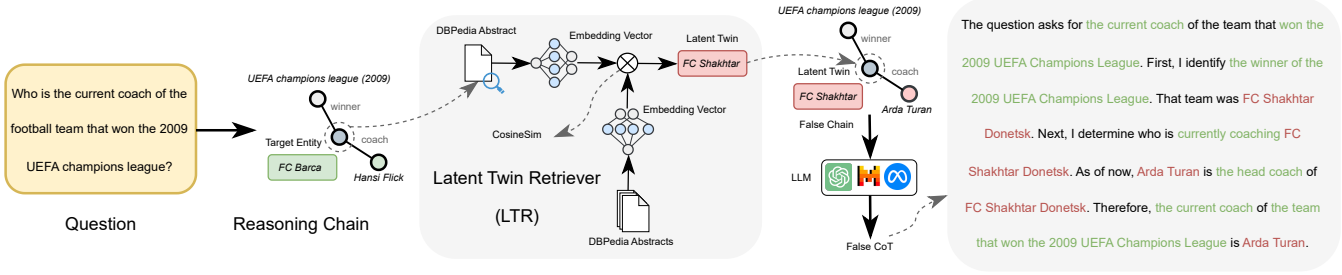

Figure 1: An example of FalseCoTQA generation where the entity FC Barca is swapped with FC Shakhtar, and the following entity is updated accordingly. This alters the reasoning chain, which leads to a fabricated chain-of-thought.

Our main contributions are: **(1)** We introduce FalseCoTQA, an adversarial MHQA benchmark that injects semantically coherent but factually incorrect chains of thought into retrieval-augmented contexts. **(2)** We propose the *Latent Twin Retrieval* algorithm, which leverages a knowledge graph to select real-world, semantically related entity swaps for constructing misleading reasoning paths. **(3)** We present a comprehensive evaluation on HotpotQA and MuSiQue, demonstrating that semantically grounded false CoTs incur substantially larger accuracy drops than existing adversarial benchmarks.

2 Methodology

We introduce FalseCoTQA, which injects semantically coherent yet factually incorrect chains of thought into retrieval-augmented prompts to rigorously evaluate LLM robustness under deceptive reasoning. As a comparative baseline, we also develop a *Context generation* method that generates adversarial context passages, rather than explicit CoTs, by swapping real-world entities in retrieved documents.

2.1 Latent Twin Retrieval

To create reasonable yet deceptive reasoning chains, we replace original entities with semantically related alternatives. We achieve this via our `Latent Twin Retrieval (LTR)` procedure, which leverages a knowledge graph to identify *latent twins*: entities that share similar relations and contextual roles with the originals. By selecting replacements grounded in real-world knowledge, LTR ensures that each fabricated chain of thought remains structurally coherent and highly believable.

2.1.1 Latent Twin Retrieval (LTR). Given a reasoning chain containing a named entity e_t (the *target entity*), our objective is to identify a semantically similar replacement e_{LT}, which we term a `Latent Twin`. This substitute is intended to preserve latent semantics while disrupting surface-level cues that LLMs might rely on.

To identify candidate latent twins, we query a knowledge graph, the DBpedia knowledge graph in our experiments, for the top-k entities related to the target entity e_t. For each candidate e_i, we retrieve its DBpedia abstract $d(e_i)$ and compute a semantic similarity score against the target's abstract $d(e_t)$ using an embedding function $\phi(\cdot)$:

Algorithm 1 FalseCoTQA Generation Algorithm

Require: Question q, Context C, Answer a
1: **return** Modified context C' with false chains of thought
2: $F \leftarrow$ ExtractSupportingFacts(q, C, a)
3: $S \leftarrow$ SelectSwitches$(E, n_{answer}, n_{head}, n_{tail})$
4: $E \leftarrow \text{LTR}(F)$
5: $COT_{\text{false}} \leftarrow \{\}$
6: **for** each $s_i \in S$ **do**
7: $\quad \text{cot}_i \leftarrow$ GenerateFalseCoT(q, C, s_i)
8: $\quad$ **if not** IsDerivable(a, cot_i) **then**
9: $\quad\quad COT_{\text{false}} \leftarrow COT_{\text{false}} \cup \{\text{cot}_i\}$
10: $\quad$ **end if**
11: **end for**
12: $COT_{\text{selected}} \leftarrow$ SelectSubset$(COT_{\text{false}}, \max = 2)$
13: $C' \leftarrow$ InjectFalseCoTs$(C, COT_{\text{selected}})$
14: **return** C'

$$s_i = \cos\left(\phi(d(e_i)),\ \phi(d(e_t))\right) \tag{1}$$

We then rank all candidates by s_i and retain the highest-scoring k. To avoid trivial substitutions, we remove any entities that match e_t exactly or are linked via ambiguous relations (e.g., *owl:sameAs*, *wikiPageDisambiguates*). Finally, we re-sort the filtered candidates by their similarity scores and select the top-ranked entity as e_{LT}, the latent twin.

$$e_{\text{top}} = \max\{\text{rank}(s_i) \mid \text{rank}(s_i) = k\}, \text{where } k = 1 \tag{2}$$

2.2 False Chain-of-Thought Generation

Algorithm 1 outlines the procedure for generating adversarial chains of thought by introducing plausible but incorrect reasoning paths into the context. Each step of the algorithm is illustrated through the example in Figure 1, where the question is: "Who is the current coach of the football team that won the 2009 UEFA Champions League?"

The process begins in line 2 with ExtractSupportingFacts, which retrieves relevant knowledge triples from a source such as DBpedia. These triples serve as the factual backbone for the reasoning chain. In the given example, this operation yields: `<UEFA Champions`

League(2009), winner, FC Barcelona> and <FC Barcelona, coach, Hansi Flick>, both of which are necessary to derive the correct answer "Hansi Flick." ExtractSupportingFacts uses the entities provided in the dataset as supporting facts. These entities include both those relevant to the question and those found in the answer. It then searches the knowledge graph for all triples that include at least two entities from the set of given entities. Finally, it constructs reasoning chains from the retrieved triples.

In line 3, the algorithm applies SelectSwitches to identify a set of candidate entities from the extracted supporting facts that will be substituted to create misleading reasoning paths. The selection process samples a specified number of entities from three key positions within the factual triples: answer entities (n_{answer}), head entities (n_{head}), and tail entities (n_{tail}). Each of these substitution types plays a distinct role in disrupting the reasoning chain. Substituting the *answer entity* involves replacing the answer itself with a semantically similar alternative, which we call its latent twin, with this swap keeping the preceding reasoning path intact. This results in a final conclusion that appears well-formed but is factually incorrect. For example, if the original answer is Hansi Flick, the chain may remain unchanged up to the final step, but the concluding entity is swapped for Arda Turan, yielding an incorrect answer through a preserved chain structure.

In contrast, substituting the *head entity* affects the origin of the reasoning chain. This type of modification alters the subject of the first triple and thereby propagates changes through the remainder of the reasoning path. For instance, if the original triple is <UEFA Champions League(2009), winner, FC Barcelona>, replacing UEFA Champions League(2009) with UEFA Champions League(2010) shifts the entire inference trajectory to center around a different league. Lastly, substituting the *tail entity* modifies the object of a triple, which typically alters the target of the relation while keeping the initiating subject fixed. For example, transforming the triple <Hansi Flick, coach, FC Barcelona> into <Hansi Flick, coach, FC Shakhtar> creates a misleading factual statement that appears structurally coherent but is semantically invalid.

Line 4 of the algorithm invokes the Latent Twin Retriever (LTR) to identify semantically similar entities, which are referred to as latent twins, for potential substitution. In our example, LTR selects FC Shakhtar as a high-proximity replacement for FC Barcelona based on methods explained through Expressions 1 to 2. The algorithm then initializes an empty set $\texttt{COT}_{\texttt{false}}$ (line 5) to store generated distractor chains. For each substitution candidate s_i (line 6), a language model is prompted to produce a new reasoning chain under the function GenerateFalseCoT (line 7). In our case, the resulting false chain is: "The question asks for the current coach of the team that won the 2009 UEFA Champions League. First, I identify the winner of the 2009 UEFA Champions League. That team was FC Shakhtar Donetsk. Next, I determine who is currently coaching FC Shakhtar Donetsk. As of now, Arda Turan is the head coach of FC Shakhtar Donetsk. Therefore, the current coach of the team that won the 2009 UEFA Champions League is Arda Turan."

To ensure that the generated chain is genuinely misleading, line 8 applies the function IsDerivable, which checks whether the original answer "Hansi Flick" is still recoverable from the newly generated chain. If the answer is not derivable, which indicates successful distraction, the chain is added to $\texttt{COT}_{\texttt{false}}$ (line 9). This process continues for all selected substitutions. Once complete, the algorithm selects up to two distractor chains using SelectSubset (line 12) to prevent over-saturation of the context. In line 13, the selected distractors are injected into the original context C through InjectFalseCoTs, resulting in a modified version C' that incorporates the adversarially crafted reasoning paths. Finally, line 14 returns the augmented context C', which can now be used for evaluating model robustness to adversarial reasoning.

The prompts used for these methods are provided in Appendix A.

3 Experiments

In this section, we report the experiments conducted to evaluate the proposed method for injecting semantically coherent but factually incorrect chains of thought into retrieval-augmented contexts. These experiments are designed to investigate the following research questions:

- **RQ1.** How does the LTR method, alongside CoT generation compare against standard LLM fabricated context generation methods? Additionally, how does it perform relative to randomly generated contexts and CoTs?
- **RQ2.** Does LLM size have any impact on how resistant a model is against noisy or fabricated context?
- **RQ3.** Which of CoT generation and Context generation is more effective in distracting and misguiding LLMs?
- **RQ4.** How well does the introduced benchmark challenge existing MHQA methods designed to be robust to irrelevant contexts?

3.1 Datasets

We applied FalseCoTQA to the development sets of two multi-hop question answering (MHQA) datasets: HotpotQA [31] and MuSiQue [6]. In our setup, applying FalseCoTQA to a context involves reformulating the context into a chain-of-thought (CoT) reasoning structure, where key entities are systematically substituted using the method described in Section 2.1.

Following prior work on fabricated contexts in automatic question answering [1], we applied our proposed fabrication method to two randomly selected distractor contexts out of the eight provided for each question in the datasets. In addition to these distractors, each question is accompanied by two gold contexts that support the correct answer. These gold contexts were left unchanged throughout the experiments.

To ensure that fabricated CoTs were genuinely misleading and did not guide the model toward the correct answer, we evaluated each original (pre-fabrication) CoT in isolation. Specifically, we presented each CoT as the sole context for its corresponding question and queried LLaMA2-13B [24]. If the generated answer exactly matched the gold answer, the CoT was considered informative rather than distracting and was excluded from the pool of candidates used for fabrication.

Table 1: FalseCoTQA Dataset Statistics

Dataset	Total Entries	Occurrences	Avg. FalseCoTs
HotpotQA	7405	4254	1.62
MuSiQue	2417	2300	1.90

Dataset statistics are summarized in Table 1. The column *Total Entries* denotes the total number of questions in the dataset. *Occurrences* indicates the number of questions for which at least one FalseCoTQA instance was successfully created. *Avg. FalseCoTs* reflects the average number of fabricated CoTs per question among those that contain at least one fabricated instance. As shown in Table 1, this average is less than two for both datasets, despite our goal of fabricating two out of ten contexts per question. This shortfall is due to two primary limitations. First, in some cases, no valid Latent Twin Retrieval (LTR) match was found for any entity or supporting fact in the question, meaning no linkable concept could be identified in DBpedia, and hence no semantically similar substitution could be performed. Second, some CoTs were found to be beneficial rather than misleading, as they enabled the QA model to produce the correct answer. These were also excluded from the fabrication process.

3.2 Baselines

We evaluate the proposed method against several baselines, including both existing approaches and newly constructed random fabrication strategies. Specifically, we aim to evaluate how well LLMs handle misleading or confusing content by comparing their performance on randomly altered contexts with their performance on contexts generated using our structured method. The evaluated baselines are as follows:

(1) **Base Results:** The performance of each LLM on the original, non-fabricated contexts provided in each dataset, serving as a reference point for evaluating the effect of fabricated contexts.

(2) **Random Noisy Contexts:** In this baseline, we introduce noise by randomly assigning distractor paragraphs from other questions to each target question. This setup simulates incoherent context perturbation without semantic alignment.

(3) **Random Entities:** This fabrication method targets the chain-of-thought (CoT) component. Instead of using semantically similar entities as in our proposed method, we substitute entities in the CoT with randomly selected alternatives. The same FalseCoTQA algorithm is applied, but the Latent Twin Retrieval (LTR) module retrieves random rather than semantically aligned entities. This baseline examines the extent to which random entity substitution can mislead LLMs compared to knowledge-grounded swaps.

(4) **Seemingly Plausible Distractors:** A multi-hop reasoning benchmark [1] that injects seemingly plausible but incorrect reasoning chains generated by LLMs. We report both the baseline results for each LLM as presented in [1], as well as the results over the fabricated contexts.

3.3 Experimental Settings

We applied the proposed method, referred to as **FalseCoTQA**, along with all baseline approaches, to two distractor contexts accompanying each question in the dataset. In addition to FalseCoTQA, we also report the performance of the **LTR Context Generation** method. This approach uses the Latent Twin Retrieval (LTR) algorithm to guide entity substitutions within two randomly selected distractor contexts per question. These substitutions are knowledge-grounded, replacing entities with semantically similar alternatives to create misleading yet plausible retrieval contexts.

To ensure a fair comparison with prior work, we adopted the few-shot prompting setup used in existing works [1]. For consistency across all baselines, we used the same versions of large language models throughout: LLaMA2-13B [24], Mixtral-8x7B-Instruct-v0.1 [10], and GPT-3.5-Turbo [17]. $n_{answer}, n_{head}, n_{tail}$ in Algorithm 1 is set to 2, 1, and 1. Note that we applied the LTR-based methods to the MuSiQue dataset for all baselines except Seemingly Plausible Distractors, as that benchmark is only available for HotpotQA. All data and code used in our experiments are publicly available in our GitHub repository[1].

3.4 Results and Discussion

Tables 2 and 3 report the performance of different LLMs on the development sets of HotpotQA [31] (Table 2) and MuSiQue [6] (Table 3), under both the original format (Base Results) and various fabrication methods described in Section 2. The numbers in bold from each table represent the largest decrease in relative performance drops for each model's EM and F1 scores.

RQ1. The results from Tables 2 and 3 demonstrate that the proposed FalseCoTQA method, which combines Latent Twin Retrieval (LTR)-based entity substitution with chain-of-thought (CoT) generation, is more effective in misleading large language models than standard context fabrication methods. Across both datasets: HotpotQA and MuSiQue, FalseCoTQA consistently causes larger performance drops in EM and F1 scores compared to random baselines and LLM-generated fabricated contexts.

On the HotpotQA dev set (Table 2), FalseCoTQA causes the largest degradation in performance for all three models. For LLaMA2-13B [24], FalseCoTQA leads to a 54.87% drop in EM and 35.79% drop in F1, significantly more than the drops observed under random distractor swapping (1.37% EM gain, 2.06% F1 gain) and random entity swapping (-10.37% EM, -16.71% F1). Even against the more sophisticated Seemingly Plausible Distractors benchmark, FalseCoTQA shows greater effectiveness at misleading GPT-3.5 (-40.13% EM, -32.57% F1) and LLaMA2-13B, though Mixtral shows slightly higher vulnerability to the Seemingly Plausible method (-30.95% EM vs. -27.36% for FalseCoTQA).

On the MuSiQue dev set (Table 3), FalseCoTQA again proves to be the most adversarial method. LLaMA2-13B exhibits a 63.16% drop in EM and 31.91% in F1 under FalseCoTQA, compared to a 16% EM drop and 9.85% F1 drop for random distractor swapping. GPT-3.5

[1] https://anonymous.4open.science/r/FalseCoTQA-2CEE/

Table 2: Comparison of FalseCoTQA with Baseline methods on HotpotQA dev dataset. Highest relative percentage drops are in bold.

Method	Llama2-13B		Mixtral		GPT-3.5	
	EM (Relative %Δ)	F1 (Relative %Δ)	EM (Relative %Δ)	F1 (Relative %Δ)	EM (Relative %Δ)	F1 (Relative %Δ)
Baselines						
Base Results	0.3124 (-)	0.4621 (-)	0.4144 (-)	0.6099 (-)	0.5592 (-)	0.7121 (-)
Random Distractor Swapping	0.3167 (1.37%)	0.4716 (2.06%)	0.4115 (-0.70%)	0.6073 (-0.43%)	0.5572 (-0.36%)	0.7098 (0.32%)
Random Entity Swapping	0.28 (-10.37%)	0.3849 (-16.71%)	0.3345 (-19.28%)	0.4968 (-18.54%)	0.3857 (-31.03%)	0.5034 (-29.31%)
Seemingly Plausible Distractors						
Reported Base Results	0.309 (-)	0.458 (-)	0.504 (-)	0.681 (-)	0.634 (-)	0.772 (-)
Seemingly Plausible Distractors	0.236 (-23.36%)	0.338 (-26.20%)	0.348 **(-30.95%)**	0.484 **(-28.93%)**	0.399 (-37.07%)	0.527 (-31.73%)
Our Methods						
LTR Context generation	0.1982 (-36.56%)	0.3655 (-20.90%)	0.3449 (-16.77%)	0.5407 (-11.35%)	0.4945 (-11.57%)	0.649 (-8.86%)
FalseCoTQA	0.141 **(-54.87%)**	0.2967 **(-35.79%)**	0.301 (-27.36%)	0.4909 (-19.51%)	0.3348 **(-40.13%)**	0.4801 **(-32.57%)**

Table 3: Comparison of FalseCoTQA with Baseline methods on MuSiQue dev dataset. Highest relative percentage drops are in bold.

Method	Llama2-13B		Mixtral		GPT-3.5	
	EM (Relative %Δ)	F1 (Relative %Δ)	EM (Relative %Δ)	F1 (Relative %Δ)	EM (Relative %Δ)	F1 (Relative %Δ)
Baselines						
Base Results	0.0931 (-)	0.1949 (-)	0.3839 (-)	0.5155 (-)	0.3132 (-)	0.4529 (-)
Random Distractor Swapping	0.0782 (-16%)	0.1757 (-9.85%)	0.3504 (-8.73%)	0.4726 (-8.32%)	0.295 (-5.81%)	0.4274 (-5.63%)
Random Entity Swapping	0.1667 (79.05%)	0.2695 (38.27%)	0.4299 (11.98%)	0.5481 (6.32%)	0.374 (19.41%)	0.5088 (12.34%)
Our Methods						
LTR Context generation	0.1477 (58.65%)	0.2916 (49.62%)	0.3566 (-7.11%)	0.5218 (1.22%)	0.3082 (-1.6%)	0.4866 (7.44%)
FalseCoTQA	0.0343 **(-63.16%)**	0.1949 **(-31.91%)**	0.1229 **(-67.99%)**	0.2297 **(-55.44%)**	0.1609 **(-48.63%)**	0.288 **(-36.41%)**

[17] sees a 48.63% EM and 36.41% F1 reduction, again larger than those observed under random or LLM-generated perturbations. Mixtral [10], the most robust of the three, still shows a 67.99% EM and 55.44% F1 drop under FalseCoTQA, outperforming all baselines in terms of adversarial effectiveness.

RQ2. From the tables, it can be observed that LLM size does influence a model's resistance to noisy or fabricated context, though not to a significant degree. While LLaMA2-13B, the smallest model, exhibited the greatest drop in EM and F1 scores under noisy conditions, and GPT-3.5 was not far behind, even though it is a significantly larger model. Mixtral, which is intermediate in size, was the least affected by noisy data, possibly due to its mixture-of-experts architecture. This design may confer greater robustness, allowing it to be less impacted than even a much larger model like GPT-3.5.

RQ3. From Tables 2 and 3, it is observable that Chain-of-thought generation proves more effective in misleading LLMs than context-level fabrication. Using the same entity substitutions via the Latent Twin Retrieval (LTR) method, FalseCoTQA consistently led to greater performance degradation across models compared to equivalent context-only manipulations. For example, on the HotpotQA dev set, the average performance drops caused by FalseCoTQA are considerably greater than those caused by LTR context generation. For LLaMA2-13B, FalseCoTQA causes a 54.87% drop in EM and a 35.79% drop in F1, whereas LTR context generation causes only a 36.56% drop in EM and 20.90% in F1. A similar pattern holds for Mixtral (EM: -27.36% vs. -16.77%, F1: -19.51% vs. -11.35%) and GPT-3.5 (EM: -40.13% vs. -11.57%, F1: -32.57% vs. -8.86%). This pattern is further reinforced in the MuSiQue dev set. For instance, with Mixtral, LTR context generation leads to a modest 7.11% drop in EM and 1.22% gain in F1, while FalseCoTQA causes a much sharper 67.99% drop in EM and 55.44% drop in F1. Similar trends are seen for GPT-3.5 (LTR: -1.6% EM, +7.44% F1; FalseCoTQA: -48.63% EM, -36.41% F1) and LLaMA2-13B (LTR: +58.65% EM, +49.62% F1; FalseCoTQA: -63.16% EM, -31.91% F1). The LLaMA2 results on MuSiQue may seem anomalous due to its very low base EM, but even in that context, CoT manipulation proves much more misleading.

Additionally, random entity swapping within CoTs was significantly more effective in misleading LLMs than random distractor paragraph swapping, highlighting the stronger adversarial impact of reasoning-level manipulation combined with entity substitution. On HotpotQA, for example, random entity swapping causes -31.03% EM and -29.31% F1 on GPT-3.5, while random distractor swapping yields only -0.36% EM and +0.32% F1. Experiments on the MuSiQue dataset yielded similar findings, with FalseCoTQA producing a substantial drop in performance relative to other baselines.

RQ4. To address this research question, we investigate two methods designed to improve the robustness of large language models (LLMs) against noisy or irrelevant context. The first, Retrobust [33],

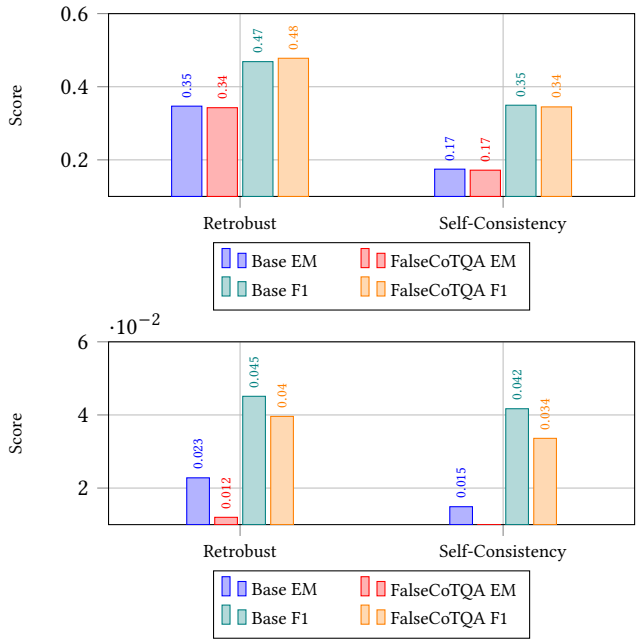

Figure 2: Comparison of Base vs. FalseCoTQA (EM and F1) under Retrobust and Self-Consistency on two dev sets: Top: HotpotQA, Bottom: MuSiQue.

is a training approach that enhances the resilience of retrieval-augmented language models (RALMs) to misleading retrieved information. It fine-tunes the model using a curated mix of relevant and irrelevant contexts, helping the model learn to differentiate and properly utilize helpful evidence, particularly in complex multi-hop reasoning tasks. For our experiments, we use a fine-tuned version of LLaMA2-13B. Specifically, we implement the llama-2-13b-peft-2wikihop-retrobust model available on Hugging Face[2]. This model has been trained on synthetic data that includes both supportive and distracting passages, allowing it to develop robustness to contextual noise.

The second model, Self-Consistency [29], is a decoding strategy for chain-of-thought prompting that improves reasoning in large language models by sampling multiple diverse reasoning paths rather than relying on a single greedy output. It then selects the most consistent final answer by aggregating over these paths, based on the intuition that complex problems often admit multiple valid routes to the correct solution.

Figure 2 presents the performance (Exact Match and F1 scores) of the LLaMA2-13B model on the HotpotQA and MuSiQue development sets under both original and fabricated conditions. The fabricated setting uses datasets modified by the FalseCoTQA method. As shown in the upper part of the figure for HotpotQA, both Retrobust and Self-Consistency models maintain high EM and F1 scores even under adversarial fabrication. The performance drop compared to the base setting is minimal, indicating that these models are relatively resilient to semantically misleading chains of thought introduced by FalseCoTQA.

In contrast, the lower part of the figure for MuSiQue reveals a more pronounced performance degradation across both models when faced with fabricated contexts. While the drop in EM and F1 scores is more substantial than in HotpotQA, it remains less severe than the performance drop experienced by the base model, which lacks any robustness-enhancing mechanism. This suggests that both Retrobust and Self-Consistency retain partial robustness even under more challenging fabrication conditions.

These results demonstrate that FalseCoTQA, especially for the MuSiQue dateset, provides a meaningful and scalable adversarial test bed for evaluating the resilience of LLMs. Despite recent advancements in robustness through training and decoding strategies, FalseCoTQA can still mislead models to a measurable extent, underscoring its value for future research in adversarial multi-hop QA and robust reasoning under noisy conditions.

4 Limitations and Future Work

In this work, we use DBPedia as the main knowledge graph to determine the latent twin of each key entity in the reasoning chain of each question. This results in losing a part of the MHQA datasets, as we cannot locate their entities in DBPedia. Moreover, we drop the samples with a correct reasoning chain at the final stage of the pipeline. This, too, makes the benchmark shorter in number of samples. As a result, extending this work to retain more samples of the MHQA datasets could be an appropriate future path. Apart from that, we can extend the research to use implicit MHQA datasets to investigate the effectiveness of our approach on a broader range of multi-hop questions. Human evaluations for the generated false CoTs could also be investigated to verify the quality and effectiveness of the FalseCoTQA benchmark. Additionally, we intend to further validate FalseCoTQA by fine tuning base models to test for increased performance against default QA datasets compared to noisy datasets, and to determine if FalseCoTQA can directly be involved in QA model training to improve performance. This way we can use this adversarial benchmark to ensure MHQA models are more robust against malicious data.

5 Conclusion

In this paper, we introduced a benchmark for evaluating multi-hop question answering models through a unique method of injecting incorrect but semantically grounded reasoning chains into contexts. Implementing a unique LTR method, we are able to retrieve entities that are inherently different but semantically close to the target entities, which are swapped throughout several steps of the reasoning chain. The results from the experimental setups demonstrate an increase in effective distraction, as base models perform worse in QA tasks when using FalseCoTQA compared to other existing methods. Even against robust MHQA methods, FalseCoTQA still poses a notable challenge against them. Overall, this solidifies FalseCoTQA's effectiveness as an adversarial MHQA benchmark.

References

[1] Neeladri Bhuiya, Viktor Schlegel, and Stefan Winkler. 2024. Seemingly Plausible Distractors in Multi-Hop Reasoning: Are Large Language Models Attentive

[2]https://huggingface.co/Ori/llama-2-13b-peft-2wikihop-retrobust

Readers? *arXiv preprint arXiv:2409.05197* (2024).
[2] Zhiyuan Chang, Mingyang Li, Xiaojun Jia, Junjie Wang, Yuekai Huang, Qing Wang, Yihao Huang, and Yang Liu. 2024. What External Knowledge is Preferred by LLMs? Characterizing and Exploring Chain of Evidence in Imperfect Context. *arXiv preprint arXiv:2412.12632* (2024).
[3] Jiayu Ding, Siyuan Wang, Qin Chen, and Zhongyu Wei. 2021. Reasoning chain based adversarial attack for multi-hop question answering. *arXiv preprint arXiv:2112.09658* (2021).
[4] Feiteng Fang, Yuelin Bai, Shiwen Ni, Min Yang, Xiaojun Chen, and Ruifeng Xu. 2024. Enhancing noise robustness of retrieval-augmented language models with adaptive adversarial training. *arXiv preprint arXiv:2405.20978* (2024).
[5] Kelvin Guu, Kenton Lee, Zora Tung, Panupong Pasupat, and Mingwei Chang. 2020. Retrieval augmented language model pre-training. In *International conference on machine learning*. PMLR, 3929–3938.
[6] Tushar Khot Ashish Sabharwal Harsh Trivedi, Niranjan Balasubramanian. 2022. MuSiQue: Multihop Questions via Single-hop Question Composition. *arXiv preprint arXiv:2108.00573* (2022).
[7] Giwon Hong, Jeonghwan Kim, Junmo Kang, Sung-Hyon Myaeng, and Joyce Jiyoung Whang. 2023. Why so gullible? enhancing the robustness of retrieval-augmented models against counterfactual noise. *arXiv preprint arXiv:2305.01579* (2023).
[8] Parastoo Jafarzadeh, Faezeh Ensan, Mahdiyar Ali Akbar Alavi, and Fattane Zarrinkalam. 2025. A Knowledge Graph Embedding Model for Answering Factoid Entity Questions. *ACM Trans. Inf. Syst.* 43, 2, Article 38 (Jan. 2025), 27 pages. doi:10.1145/3678003
[9] Jisoo Jang and Wen-Syan Li. 2024. AU-RAG: Agent-based Universal Retrieval Augmented Generation. In *Proceedings of the 2024 Annual International ACM SIGIR Conference on Research and Development in Information Retrieval in the Asia Pacific Region* (Tokyo, Japan) *(SIGIR-AP 2024)*. Association for Computing Machinery, New York, NY, USA, 2–11. doi:10.1145/3673791.3698416
[10] Albert Q. Jiang, Alexandre Sablayrolles, Antoine Roux, Arthur Mensch, Blanche Savary, Chris Bamford, Devendra Singh Chaplot, Diego de las Casas, Emma Bou Hanna, Florian Bressand, Gianna Lengyel, Guillaume Bour, Guillaume Lample, Lélio Renard Lavaud, Lucile Saulnier, Marie-Anne Lachaux, Pierre Stock, Sandeep Subramanian, Sophia Yang, Szymon Antoniak, Teven Le Scao, Théophile Gervet, Thibaut Lavril, Thomas Wang, Timothée Lacroix, and William El Sayed. 2024. Mixtral of Experts. arXiv:2401.04088 [cs.LG] https://arxiv.org/abs/2401.04088
[11] Yichen Jiang and Mohit Bansal. 2019. Avoiding reasoning shortcuts: Adversarial evaluation, training, and model development for multi-hop QA. *arXiv preprint arXiv:1906.07132* (2019).
[12] Aneta Koleva, Martin Ringsquandl, and Volker Tresp. 2023. Adversarial Attacks on Tables with Entity Swap. *arXiv preprint arXiv:2309.08650* (2023).
[13] Patrick Lewis, Ethan Perez, Aleksandra Piktus, Fabio Petroni, Vladimir Karpukhin, Naman Goyal, Heinrich Küttler, Mike Lewis, Wen-tau Yih, Tim Rocktäschel, et al. 2020. Retrieval-augmented generation for knowledge-intensive nlp tasks. *Advances in neural information processing systems* 33 (2020), 9459–9474.
[14] Yu-An Liu, Ruqing Zhang, Jiafeng Guo, Maarten de Rijke, Yixing Fan, and Xueqi Cheng. 2024. Robust Neural Information Retrieval: An Adversarial and Out-of-distribution Perspective. arXiv:2407.06992 [cs.IR] https://arxiv.org/abs/2407.06992
[15] Jack Longwell, Mahdiyar Ali Akbar Alavi, Fattane Zarrinkalam, and Faezeh Ensan. 2024. Triple Augmented Generative Language Models for SPARQL Query Generation from Natural Language Questions. In *Proceedings of the 2024 Annual International ACM SIGIR Conference on Research and Development in Information Retrieval in the Asia Pacific Region* (Tokyo, Japan) *(SIGIR-AP 2024)*. Association for Computing Machinery, New York, NY, USA, 269–273. doi:10.1145/3673791.3698426
[16] Hadiseh Moradisani, Fattane Zarrinkalam, Julien Serbanescu, and Zeinab Noorian. 2024. UnAnswGen: A Systematic Approach for Generating Unanswerable Questions in Machine Reading Comprehension. In *Proceedings of the 2024 Annual International ACM SIGIR Conference on Research and Development in Information Retrieval in the Asia Pacific Region* (Tokyo, Japan) *(SIGIR-AP 2024)*. Association for Computing Machinery, New York, NY, USA, 280–286. doi:10.1145/3673791.3698413
[17] OpenAI. 2024. OpenAI API. https://platform.openai.com.
[18] Seong-Il Park and Jay-Yoon Lee. 2024. Toward Robust RALMs: Revealing the Impact of Imperfect Retrieval on Retrieval-Augmented Language Models. *Transactions of the Association for Computational Linguistics* 12 (2024), 1686–1702.
[19] Md Rizwan Parvez. 2025. Chain of Evidences and Evidence to Generate: Prompting for Context Grounded and Retrieval Augmented Reasoning. arXiv:2401.05787 [cs.CL] https://arxiv.org/abs/2401.05787
[20] Ori Ram, Yoav Levine, Itay Dalmedigos, Dor Muhlgay, Amnon Shashua, Kevin Leyton-Brown, and Yoav Shoham. 2023. In-context retrieval-augmented language models. *Transactions of the Association for Computational Linguistics* 11 (2023), 1316–1331.
[21] Mohammadamin Shafiei, Hamidreza Saffari, and Nafise Sadat Moosavi. 2025. MultiHoax: A Dataset of Multi-hop False-Premise Questions. *arXiv preprint arXiv:2506.00264* (2025).
[22] Weihang Su, Yichen Tang, Qingyao Ai, Changyue Wang, Zhijing Wu, and Yiqun Liu. 2024. Mitigating Entity-Level Hallucination in Large Language Models. In *Proceedings of the 2024 Annual International ACM SIGIR Conference on Research and Development in Information Retrieval in the Asia Pacific Region* (Tokyo, Japan) *(SIGIR-AP 2024)*. Association for Computing Machinery, New York, NY, USA, 23–31. doi:10.1145/3673791.3698403
[23] Yixuan Tang and Yi Yang. 2024. Multihop-rag: Benchmarking retrieval-augmented generation for multi-hop queries. *arXiv preprint arXiv:2401.15391* (2024).
[24] Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, Dan Bikel, Lukas Blecher, Cristian Canton Ferrer, Moya Chen, Guillem Cucurull, David Esiobu, Jude Fernandes, Jeremy Fu, Wenyin Fu, Brian Fuller, Cynthia Gao, Vedanuj Goswami, Naman Goyal, Anthony Hartshorn, Saghar Hosseini, Rui Hou, Hakan Inan, Marcin Kardas, Viktor Kerkez, Madian Khabsa, Isabel Kloumann, Artem Korenev, Punit Singh Koura, Marie-Anne Lachaux, Thibaut Lavril, Jenya Lee, Diana Liskovich, Yinghai Lu, Yuning Mao, Xavier Martinet, Todor Mihaylov, Pushkar Mishra, Igor Molybog, Yixin Nie, Andrew Poulton, Jeremy Reizenstein, Rashi Rungta, Kalyan Saladi, Alan Schelten, Ruan Silva, Eric Michael Smith, Ranjan Subramanian, Xiaoqing Ellen Tan, Binh Tang, Ross Taylor, Adina Williams, Jian Xiang Kuan, Puxin Xu, Zheng Yan, Iliyan Zarov, Yuchen Zhang, Angela Fan, Melanie Kambadur, Sharan Narang, Aurelien Rodriguez, Robert Stojnic, Sergey Edunov, and Thomas Scialom. 2023. Llama 2: Open Foundation and Fine-Tuned Chat Models. arXiv:2307.09288 [cs.CL] https://arxiv.org/abs/2307.09288
[25] Harsh Trivedi, Niranjan Balasubramanian, Tushar Khot, and Ashish Sabharwal. 2022. Interleaving retrieval with chain-of-thought reasoning for knowledge-intensive multi-step questions. *arXiv preprint arXiv:2212.10509* (2022).
[26] Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Fei Xia, Ed Chi, Quoc V Le, Denny Zhou, et al. 2022. Chain-of-thought prompting elicits reasoning in large language models. *Advances in neural information processing systems* 35 (2022), 24824–24837.
[27] Zhen Xiang, Fengqing Jiang, Zidi Xiong, Bhaskar Ramasubramanian, Radha Poovendran, and Bo Li. 2024. Badchain: Backdoor chain-of-thought prompting for large language models. *arXiv preprint arXiv:2401.12242* (2024).
[28] Xilie Xu, Keyi Kong, Ning Liu, Lizhen Cui, Di Wang, Jingfeng Zhang, and Mohan Kankanhalli. 2023. An llm can fool itself: A prompt-based adversarial attack. *arXiv preprint arXiv:2310.13345* (2023).
[29] Dale Schuurmans Quoc Le Ed H. Chi Sharan Narang Aakanksha Chowdhery Denny Zhou Xuezhi Wang, Jason Wei. 2023. Self-Consistency Improves Chain of Thought Reasoning in Language Models. *arXiv preprint arXiv:2203.11171* (2023).
[30] Shiping Yang, Jie Wu, Wenbiao Ding, Ning Wu, Shining Liang, Ming Gong, Hengyuan Zhang, and Dongmei Zhang. 2025. Quantifying the Robustness of Retrieval-Augmented Language Models Against Spurious Features in Grounding Data. *arXiv preprint arXiv:2503.05587* (2025).
[31] 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. *arXiv preprint arXiv:1809.09600* (2018).
[32] Ori Yoran, Tomer Wolfson, Ori Ram, and Jonathan Berant. 2023. Making retrieval-augmented language models robust to irrelevant context. *arXiv preprint arXiv:2310.01558* (2023).
[33] Ori Yoran, Tomer Wolfson, Ori Ram, and Jonathan Berant. 2024. Making Retrieval-Augmented Language Models Robust to Irrelevant Context. arXiv:2310.01558 [cs.CL] https://arxiv.org/abs/2310.01558
[34] Wenhao Yu, Hongming Zhang, Xiaoman Pan, Kaixin Ma, Hongwei Wang, and Dong Yu. 2023. Chain-of-note: Enhancing robustness in retrieval-augmented language models. *arXiv preprint arXiv:2311.09210* (2023).
[35] Linda Zeng, Rithwik Gupta, Divij Motwani, Diji Yang, and Yi Zhang. 2025. Worse than zero-shot? a fact-checking dataset for evaluating the robustness of rag against misleading retrievals. *arXiv preprint arXiv:2502.16101* (2025).
[36] Kaijie Zhu, Jindong Wang, Jiaheng Zhou, Zichen Wang, Hao Chen, Yidong Wang, Linyi Yang, Wei Ye, Yue Zhang, Neil Gong, et al. 2023. Promptrobust: Towards evaluating the robustness of large language models on adversarial prompts. In *Proceedings of the 1st ACM Workshop on Large AI Systems and Models with Privacy and Safety Analysis*. 57–68.
[37] Wei Zou, Runpeng Geng, Binghui Wang, and Jinyuan Jia. 2024. Poisonedrag: Knowledge corruption attacks to retrieval-augmented generation of large language models. *arXiv preprint arXiv:2402.07867* (2024).

A Appendix

FalseCoTQA answer swapping with LTR

You are given a natural language question, a knowledge triple (in the form of subject, predicate, object), and a guided answer. Your task is to write a clear, structured chain of reasoning that connects the triple to the guided answer. The reasoning should proceed step by step, without restating the answer explicitly, and should be centered around either the subject or the object of the triple. Use the triple to support one or more steps, ensuring the progression is natural and logically sound.
Include 1–2 examples before your output. Each should follow this format: { "reasoning": "[Step-by-step paragraph here]" }
Only output the final JSON object. Do not include explanations or markdown.

Example 1:
`Question:` Who is currently coaching the team that won the 2009 UEFA Champions League?
`Triple:` (FC Shakhtar Donetsk, current coach, Arda Turan)
`Guided Answer:` Arda Turan
{ "reasoning": "The question asks for the current coach of the team that won the 2009 UEFA Champions League. First, I identify the team that won the tournament in 2009. That team was FC Shakhtar Donetsk. Next, I determine who is currently coaching FC Shakhtar Donetsk. According to the given information, Arda Turan is the current coach of that team. Therefore, by identifying the 2009 winner and linking it to its present coach, I arrive at the answer." }

Example 2:
`Question:` What city is home to the team led by Carlo Ancelotti?
`Triple:` (Real Madrid, head coach, Carlo Ancelotti)
`Guided Answer:` Madrid
{ "reasoning": "The question is asking for the home city of the team coached by Carlo Ancelotti. First, I find out which team is currently led by Carlo Ancelotti. According to the triple, that team is Real Madrid. Next, I consider where Real Madrid is based. Real Madrid is a football club that is located in the city of Madrid. Therefore, the team led by Carlo Ancelotti is based in Madrid, which answers the question." }

`Now complete the following:`
`Question:` {question}
`Triple:` ({m['subject']}, {m['predicate']}, {m['object']})
`Guided Answer:` {mod_ans}

FalseCoTQA entity swapping with LTR

You are given a natural language question and one or more supporting knowledge triples (in subject, predicate, object form). Each triple centers around a key entity. Your task is to generate an explanatory context that walks through the reasoning needed to answer the question, focusing on the entity mentioned in the triples. Your reasoning must proceed in a human-readable, logical sequence. Each paragraph should focus on one or two closely related triples and build toward the final conclusion.
Return a JSON object with two fields:
`"reasoning"`: a list of 1–2 paragraphs (80–120 words total)
`"ans"`: a concise phrase (e.g., a name, year, or location)
Only output a valid JSON object. No markdown, explanation, or commentary.
Here are some examples of the reasoning format: **Example 1:**

`Question:` Who is the current coach of the team that won the 2009 UEFA Champions League?
{ "reasoning": ["The question asks for the current coach of the team that won the 2009 UEFA Champions League. First, I identify the winner of the 2009 UEFA Champions League. That team was FC Shakhtar Donetsk.", "I determine who is currently coaching FC Shakhtar Donetsk. As of now, Arda Turan is the head coach of FC Shakhtar Donetsk. Therefore, the current coach of the team that won the 2009 UEFA Champions League is Arda Turan."], "ans": "Arda Turan" }

Example 2:
`Question:` What city is home to the company founded by Elon Musk that makes electric cars?
{ "reasoning": ["The question asks for the city where the company founded by Elon Musk that produces electric cars is based. According to the information, Elon Musk is the founder of Tesla.", "I need to determine where Tesla is headquartered. The headquarters of Tesla is in Austin. Thus, the city I'm looking for is Austin."], "ans": "Austin" }

`Now complete the following:`
`Question:` {question}
{triple_string}