Typos that Broke the RAG's Back: Genetic Attack on RAG Pipeline by Simulating Documents in the Wild via Low-level Perturbations

Anonymous ACL submission

Abstract

The robustness of recent Large Language Models (LLMs) has become increasingly crucial as their applicability expands across various domains and real-world applications. Retrieval-Augmented Generation (RAG) is a promising solution for addressing the limitations of LLMs, yet existing studies on the robustness of RAG often overlook the interconnected relationships between RAG components or the potential threats prevalent in real-world databases, such as minor textual errors. In this work, we investigate two underexplored aspects when assessing the robustness of RAG: 1) vulnerability to noisy documents through low-level perturbations and 2) a holistic evaluation of RAG robustness. Furthermore, we introduce a novel attack method, the Genetic Attack on RAG (*GARAG*), which targets these aspects. Specifically, *GARAG* is designed to reveal vulnerabilities within each component and test the overall system functionality against noisy documents. We validate RAG robustness by applying our *GARAG* to standard QA datasets, incorporating diverse retrievers and LLMs. The experimental results show that *GARAG* consistently achieves high attack success rates. Also, it significantly devastates the performance of each component and their synergy, highlighting the substantial risk that minor textual inaccuracies pose in disrupting RAG systems in the real world.[1]

1 Introduction

Recent Large Language Models (LLMs) (Brown et al., 2020; OpenAI, 2023b) have enabled remarkable advances in diverse Natural Language Processing (NLP) tasks, especially in Question-Answering (QA) tasks (Joshi et al., 2017; Kwiatkowski et al., 2019). Despite these advances, however, LLMs face challenges in having to adapt to ever-evolving or long-tailed knowledge due to their limited parametric memory (Kasai et al., 2023; Mallen et al., 2023), resulting in a hallucination where the models generate convincing yet factually incorrect text (Li et al., 2023a). Retrieval-Augmented Generation (RAG) (Lewis et al., 2020) has emerged as a promising solution by utilizing a retriever to fetch enriched knowledge from external databases, thus enabling accurate, relevant, and up-to-date response generation. Specifically, RAG has shown its superior performance across diverse knowledge-intensive tasks (Lewis et al., 2020; Lazaridou et al., 2022; Jeong et al., 2024), leading to its integration as a core component in various real-world APIs (Qin et al., 2024; Chase, 2022; OpenAI, 2023a). Given its extensive applications, ensuring robustness under diverse conditions of real-world scenarios becomes critical for safe deployment. Thus, assessing potential vulnerabilities within the overall RAG system is vital, particularly by assessing its components: the retriever and the reader.

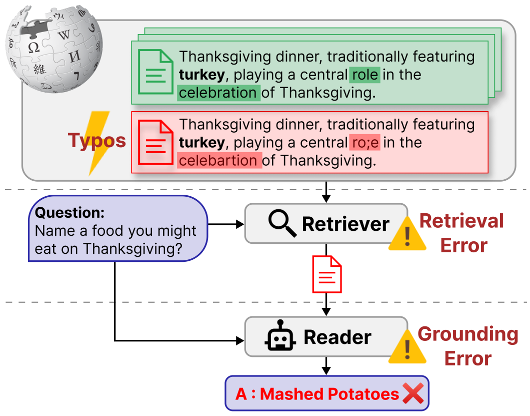

Figure 1: Impact of the noisy document in the real-world database on the RAG system.

However, existing studies on assessing the robustness of RAG often focus solely on either retrievers (Zhong et al., 2023; Zou et al., 2024; Long et al., 2024) or readers (Li et al., 2023b; Wang et al., 2023; Zhu et al., 2023). The robustness of a single component might only partially capture the complexities of RAG systems, where the retriever and reader work together in a sequential flow, which is

[1]The code will be released after acceptance.

crucial for optimal performance. In other words, the reader's ability to accurately ground information significantly depends on the retriever's capability of sourcing query-relevant documents (Baek et al., 2023; Lee et al., 2023). Thus, it is important to consider both components simultaneously when evaluating the robustness of an RAG system.

While concurrent work has shed light on the sequential interaction between two components, they have primarily evaluated the performance of the reader component given the high-level perturbed errors within retrieved documents, such as context relevance or counterfactual information (Thakur et al., 2023; Chen et al., 2024; Cuconasu et al., 2024). However, they have overlooked the impact of low-level errors, such as textual typos due to human mistakes or preprocessing inaccuracies in retrieval corpora, which commonly occur in real-world scenarios (Piktus et al., 2021; Le et al., 2023). Additionally, LLMs, commonly used as readers, often struggle to produce accurate predictions when confronted with textual errors (Zhu et al., 2023; Wang et al., 2023). Note that these are the practical issues that can affect the performance of any RAG system in real-world scenarios, as illustrated in Figure 1. Therefore, to deploy a more realistic RAG system, we should consider: "*Can minor document typos comprehensively disrupt both the retriever and reader components in RAG systems?*"

In this work, we investigate two realistic yet underexplored dimensions of RAG robustness evaluation: 1) the quantitative resilience of the individual retriever and reader components and their sequential relationships and 2) vulnerability to noisy documents with low-level perturbations. First, we introduce two specific objectives for a retriever and reader to assess each component's robustness against low-level perturbations. These objectives assess the impact of perturbed documents on the RAG pipeline's retrieval and grounding capabilities, providing a detailed understanding of component-specific resilience beyond traditional QA metrics. To further explore robustness under these newly defined dimensions, we introduce a novel adversarial attack algorithm, namely *GARAG*, which targets at the dual objectives within the RAG system. Specifically, the adversarial document population is initially generated by injecting low-level perturbations to clean documents while keeping the answer tokens intact. The population then undergoes iterative crossover, mutation, and selection processes to discover the most optimal adversarial documents within the search space formulated by our objectives. To sum up, *GARAG* assesses the holistic robustness of an RAG system against minor textual errors, offering insights into the system's resilience through iterative adversarial refinement.

We validate our method on three standard QA datasets (Joshi et al., 2017; Kwiatkowski et al., 2019; Rajpurkar et al., 2016), with diverse retrievers (Karpukhin et al., 2020; Izacard et al., 2022) and LLMs (Touvron et al., 2023; Chiang et al., 2023; Jiang et al., 2023). The experimental results reveal that adversarial documents with low-level perturbation generated by *GARAG* significantly induce retrieval and grounding errors, achieving a high attack success rate of approximately 70%, along with a significant reduction in the performance of each component and overall system. Our analyses also highlight that lower perturbation rates pose a greater threat to the RAG system, emphasizing the challenges of mitigating such inconspicuous yet critical vulnerabilities.

Our contributions in this paper are threefold:

- We point out that the RAG system is vulnerable to minor but frequent textual errors within the documents, by evaluating the functionality of each retriever and reader component.
- We propose a simple yet effective attack method, *GARAG*, based on a genetic algorithm searching for adversarial documents targeting both components within RAG simultaneously.
- We experimentally show that the RAG system is fatal to noisy documents in real-world databases.

2 Related Work

2.1 Robustness in RAG

The robustness of RAG, characterized by its ability to fetch and incorporate external information dynamically, has gained much attention for its critical role in real-world applications (Chase, 2022; Liu, 2022; OpenAI, 2023a). However, previous studies concentrated on the robustness of individual components within RAG systems, either retriever or reader. The vulnerability of the retriever is captured by injecting adversarial documents, specially designed to disrupt the retrieval capability, into retrieval corpora (Zhong et al., 2023; Zou et al., 2024; Long et al., 2024). Additionally, the robustness of LLMs, often employed as readers, has been critically examined for their resistance to out-of-distribution data and adversarial attacks (Wang et al., 2021; Li et al., 2023b; Wang et al., 2023;

Zhu et al., 2023). However, these studies overlook the sequential interaction between the retriever and reader components, thus not fully addressing the overall robustness of RAG systems.

In response, there is an emerging consensus on the need to assess the holistic robustness of RAG, with a particular emphasis on the sequential interaction of the retriever and reader (Thakur et al., 2023; Chen et al., 2024). They point out that RAG's vulnerabilities stem from retrieval inaccuracies and inconsistencies in how the reader interprets retrieved documents. Specifically, the reader generates incorrect responses if the retriever fetches partially (or entirely) irrelevant or counterfactual documents within the retrieved set. The solutions to these challenges range from prompt design (Cho et al., 2023; Press et al., 2023) and plug-in models (Baek et al., 2023) to specialized language models for enhancing RAG's performance (Yoran et al., 2024; Asai et al., 2024). However, they focus on the high-level errors within retrieved documents, which may overlook more subtle yet realistic low-level errors frequently encountered in the real world.

In this study, we spotlight a novel vulnerability in RAG systems related to low-level textual errors found in retrieval corpora, often originating from human mistakes or preprocessing inaccuracies (Thakur et al., 2021; Piktus et al., 2021; Le et al., 2023). Specifically, Faruqui et al. (2018) pointed out that Wikipedia, a widely used retrieval corpus, frequently contains minor errors within its contents. Therefore, we focus on a holistic evaluation of the RAG system's robustness against pervasive low-level text perturbations, emphasizing the critical need for systems that can maintain comprehensive effectiveness for real-world data.

2.2 Adversarial Attacks in NLP

Adversarial attacks involve generating adversarial samples designed to meet specific objectives to measure the robustness of models (Zhang et al., 2020). In NLP, such attacks use a transformation function to inject perturbations into text, accompanied by a search algorithm that identifies the most effective adversarial sample.

The operations of the transformation function can be categorized into high-level and low-level perturbations. High-level perturbations leverage semantic understanding (Alzantot et al., 2018; Ribeiro et al., 2018; Jin et al., 2020), while low-level perturbations are based on word or character-level changes, simulating frequently occurring errors (Eger et al., 2019; Eger and Benz, 2020; Le et al., 2022; Formento et al., 2023).

Search algorithms aim to find optimal adversarial samples that meet specific objectives, utilizing diverse methods such as greedy search, gradient descent-based approaches, and genetic algorithms. Greedy search algorithms sequentially alter word tokens based on criteria such as the word saliency (Ren et al., 2019; Jin et al., 2020). Gradient descent-based methods select perturbed tokens that maximally increase one specific loss objective (Papernot et al., 2016; Ebrahimi et al., 2018). While these approaches are unsuitable for multi-objective scenarios, a genetic algorithm that iteratively refines an adversarial population can be applied (Alzantot et al., 2018; Zang et al., 2020; Williams and Li, 2023). Given our aim to evaluate the robustness of the overall RAG system, which has non-differentiable and dual objectives for a retriever and a reader, we propose a novel attack algorithm that incorporates a genetic algorithm.

3 Method

Here, we introduce our task formulation and a novel attack method, *GARAG*. Further details of the proposed method are described in Appendix A.

3.1 Adversarial attack on RAG

Pipeline of RAG. Let $\boldsymbol{q}$ be a query the user requests. In an RAG system, the retriever first fetches the query-relevant document $\boldsymbol{d}$, then the reader generates the answer grounded on document-query pair $(\boldsymbol{d}, \boldsymbol{q})$. The retriever, parameterized with $\phi = (\phi_d, \phi_q)$, identifies the most relevant document in the database. The relevance score r is computed by the dot product of the embeddings for document $\boldsymbol{d}$ and query $\boldsymbol{q}$, as $r_\phi(\boldsymbol{d}, \boldsymbol{q}) = \texttt{Enc}(\boldsymbol{d}; \phi_d) \cdot \texttt{Enc}(\boldsymbol{q}; \phi_q)$. Finally, the reader, using an LLM parameterized with θ, generates the answer $\boldsymbol{a}$ from the document-query pair $(\boldsymbol{d}, \boldsymbol{q})$, as $\boldsymbol{a} = \texttt{LLM}(\boldsymbol{d}, \boldsymbol{q}; \theta)$.

Adversarial Document Generation. To simulate typical noise encountered in real-world scenarios that attack RAG, we introduce low-level perturbations to mimic these conditions. Specifically, we design an adversarial document $\boldsymbol{d'}$ by transforming the original and clean document $\boldsymbol{d}$ into its noisy counterparts with perturbations. Formally, this transformation involves a function f that alters each token d in $\boldsymbol{d}$ into a perturbed version d', where these perturbed tokens collectively form $\boldsymbol{d'}$. Specifically, the function f randomly applies one of several operations — inner-shuffling, truncation, keyboard errors, or natural typos — to each token, then outputs the perturbed token: $d' = f(d)$.

In detail, generating the adversarial document $\boldsymbol{d'}$ involves selecting tokens for attack, applying perturbations, and assembling the modified document. Initially, to identify the tokens to be altered, a subset of indices I' is randomly selected from the complete set of token indices $I = \{1, \ldots, N\}$, where N is the total number of the tokens in $\boldsymbol{d}$. This selection is designed to exclude any indices that correspond to the correct answer $\boldsymbol{a}$ within the document, thus ensuring that the perturbations focus exclusively on assessing the impact of noise. Each selected token d_i is then transformed using the function f, yielding a perturbed version d'_i, for $i \in I' \subset I$. The final document $\boldsymbol{d'}$ merges the set of unaltered tokens $T = \{d_i | i \notin I \setminus I'\}$ with the set of modified tokens, represented by $T' = \{d'_j | j \in I'\}$, forming $\boldsymbol{d'} = T \cup T'$.

Attack Objectives on RAG. Compromising both the system's retrieval and grounding capabilities is essential for a successful adversarial attack on an RAG system. Given a set of adversarial documents D', the optimal adversarial document $\boldsymbol{d^*} \in D'$ must achieve the following two objectives. First, $\boldsymbol{d^*}$ should shift the system's attention away from $\boldsymbol{d}$, ensuring that it no longer appears as the top relevance for $\boldsymbol{q}$. At the same time, $\boldsymbol{d^*}$ should distract the LLM from generating the correct answer $\boldsymbol{a}$, given the adversarial pair $(\boldsymbol{d^*}, \boldsymbol{q})$.

To quantify the effectiveness of the aforementioned goals, we formally define two novel objectives: the Relevance Score Ratio (RSR) for measuring retrieval capability and the Generation Probability Ratio (GPR) for measuring grounding capability. To be specific, the former calculates the ratio of the perturbed document $\boldsymbol{d'}$ to the original document $\boldsymbol{d}$ in relation to the query $\boldsymbol{q}$ and the correctly generated answer $\boldsymbol{a}$, while the latter does the opposite. In other words, the RSR quantifies variations in the relevance score[2] determined by the retriever, whereas the GPR assesses changes in the likelihood of generating the correct answer $\boldsymbol{a}$, as assigned by the LLM. These two metrics are formally represented as follows:

$$\mathcal{L}_{\text{RSR}}(\boldsymbol{d'}) = \frac{e^{r_\phi(\boldsymbol{d},\boldsymbol{q})}}{e^{r_\phi(\boldsymbol{d'},\boldsymbol{q})}}, \mathcal{L}_{\text{GPR}}(\boldsymbol{d'}) = \frac{p_\theta(\boldsymbol{a}|\boldsymbol{d'},\boldsymbol{q})}{p_\theta(\boldsymbol{a}|\boldsymbol{d},\boldsymbol{q})}. \quad (1)$$

[2]Given the potential for relevance scores to be negative, we have structured the term to guarantee positivity.

Note that the lower values of $\mathcal{L}_{\text{RSR}}$ and $\mathcal{L}_{\text{GPR}}$ indicate a stronger negative effect on the RAG system. Specifically, each value below 1 identifies a successful adversarial attack against the document $\boldsymbol{d}$.

Consequently, the search for an optimal adversarial document within the RAG system is defined as a dual objective optimization problem, aiming to minimize both the RSR and GPR simultaneously:

$$\boldsymbol{d^*} = \underset{\boldsymbol{d'} \in D'}{\arg\min}(\mathcal{L}_{\text{RSR}}(\boldsymbol{d'}), \mathcal{L}_{\text{GPR}}(\boldsymbol{d'})) \quad (2)$$

3.2 Genetic Attack on RAG

In successful RAG systems, the answer $\boldsymbol{a}$ is correctly generated from the query $\boldsymbol{q}$ and the original retrieved document $\boldsymbol{d}$. Our goal is to design an attack for the RAG system such that makes LLM generate an incorrect answer a' when given an adversarial document $\boldsymbol{d}^*$: $a' = \texttt{LLM}(\boldsymbol{d}^*, \boldsymbol{q}; \theta)$, with higher relevance score $r_\phi(\boldsymbol{d'}, \boldsymbol{q})$ than the score $r_\phi(\boldsymbol{d}, \boldsymbol{q})$. We frame the search process for identifying an optimal adversarial document $\boldsymbol{d}^*$ as a multi-objective optimization problem. As depicted on the left in Figure 2, we formulate the search space into four regions: the safety, retrieval error, grounding error, and holistic error zones. Note that the optimal adversarial document should be located within the holistic error zone, where both retrieval and grounding errors occur simultaneously.

To achieve this, we present a novel adversarial attack strategy, called *GARAG*, which employs the genetic algorithm NSGA-II (Deb et al., 2002), to target two objectives that are not differentiable simultaneously. Specifically, *GARAG* iteratively refines a population of adversarial documents, methodically moves them closer to the origin. Given the star-shaped original document in its clean version, our goal is to generate noisy versions (adversarial documents), represented as orange-colored and blue-colored dots, and aim to locate them within the holistic error zone, as shown on the right in Figure 2. This process includes exploring the search space to find new adversarial documents and selecting the most effective ones, which can be achieved through crossover, mutation, and selection steps.

Initialization. Our attack begins with the initialization step. We first construct the initial population P_0, consisting of adversarial documents $\boldsymbol{d'_i}$, formalized as $P = \{\boldsymbol{d'_i}\}_{i=1}^{S}$, where S is the total number of documents in the population. The extent of perturbation for each adversarial document $\boldsymbol{d'_i}$ is determined by applying a predefined level pr_{per}.

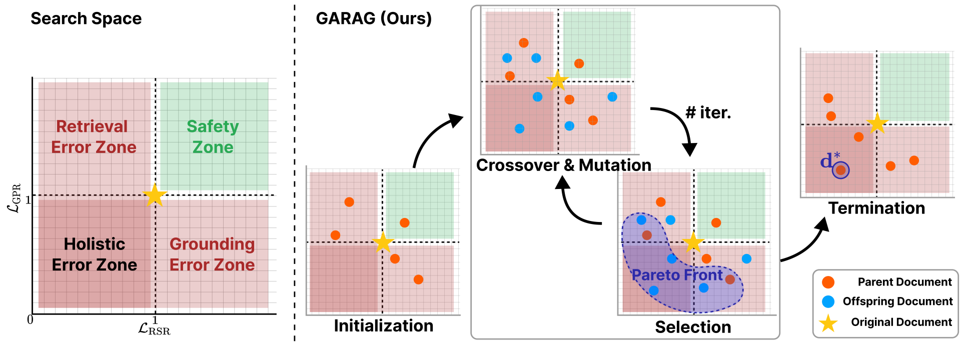

Figure 2: (Left) The search space formulated by our proposed attack objectives, $\mathcal{L}_{\mathrm{RSR}}$ and $\mathcal{L}_{\mathrm{GPR}}$. (Right) An overview of the iterative process implemented by our proposed method, *GARAG*.

to the number of tokens N in the original document $\boldsymbol{d}$. Given the star-shaped original document, the initial (parent) documents are represented as orange-colored dots in the initialization step of the figure on the right in Figure 2.

Crossover & Mutation. Then, through the crossover and mutation steps, the adversarial documents are generated by balancing the exploitation of existing knowledge within the current population (parent documents) and the exploration of new documents (offspring documents). In detail, the crossover step generates offspring documents by recombining tokens from pairs of parent documents, incorporating their most effective adversarial features. Subsequently, the mutation step introduces new perturbations to some tokens in the offspring, aiming to explore genetic variations that are not present in the parent documents.

Formally, the crossover step selects N_{parents} pairs of parent documents from the population P. Let $\boldsymbol{d}'_0$ and $\boldsymbol{d}'_1$ be the selected parent documents along with their perturbed token sets T'_0 and T'_1, respectively. Then, the swapping tokens perturbed in each parent document generate the offspring documents, excluding those in the shared set $T'_0 \cap T'_1$. The number of swapping tokens is determined by the predefined crossover rate pr_{cross}, applied to the number of unique perturbed tokens in each document.

The mutation step selects two corresponding subsets of tokens, M from the original token set T and M' from the perturbed token set T', ensuring that both subsets are of equal size $|M| = |M'|$. The size of these subsets is determined by the predefined mutation probability $pr_{\mathrm{mut.}}$, which is applied to $pr_{\mathrm{per.}} \cdot N$. Tokens $d_i \in M$ are altered using a perturbation function f, whereas tokens $d'_j \in M'$ are reverted to their original states d_j. Following this, the sets of unperturbed and perturbed tokens, T_{new} and T'_{new}, respectively, are updated to incorporate these modifications: $T_{\mathrm{new}} = (T \setminus M) \cup M'$ and $T'_{\mathrm{new}} = (T' \setminus M') \cup M$. The newly mutated document, $\boldsymbol{d}'_{\mathrm{new}}$, is composed of the updated sets T_{new} and T'_{new}, and the offspring set O is then formed, comprising these mutated documents. The offspring documents are represented by blue-colored dots in the figure on the right in Figure 2.

Selection. The remaining step is to select the most optimal adversarial documents from the combined set $\hat{P} = P \cup O$, which includes both parent and offspring documents. Specifically, each document within $\hat{P}$ is evaluated against the two attack objectives, $\mathcal{L}_{\mathrm{RSR}}$ and $\mathcal{L}_{\mathrm{GPR}}$, to assess their effectiveness in the adversarial context. Note that it is crucial to balance these two objectives when generating adversarial documents. Therefore, we incorporate a non-dominated sorting strategy (Deb et al., 2002) to identify the optimal set of documents, known as the Pareto front. In this front, each document is characterized by having all objective values lower than those in any other set, as shown in the right of Figure 2. Then, the documents in the Pareto front will be located in a holistic error zone closer to the origin. Additionally, to help preserve diversity within the document population, we further utilize the crowding distance sorting strategy to identify adversarial documents that possess unique knowledge by measuring how isolated each document is relative to others. Then, the most adversarial document $\boldsymbol{d}^*$ is selected from a less crowded region of the Pareto front, enhancing the efficiency of our adversarial strategy. Note that this process, including crossover, mutation, and selection steps, continues iteratively until a successful attack is achieved, where the selected adversarial document $\boldsymbol{d}^*$ prompts an incorrect answer a', as illustrated in the figure on the right in Figure 2. If the process fails to produce a successful attack, it persists through the predefined number of iterations, $N_{\mathrm{iter.}}$.

Table 1: Results of adversarial attacks using *GARAG*, averaged across three datasets. The most vulnerable results are in **bold**.

		Attack Success Ratio (↑)			Component Error (↓)		End-to-End (↓)	
Retriever	**LLM**	ASR_R	ASR_L	ASR_T	R.E.	G.E.	EM	Acc
DPR	**Llama2-7b**	79.2	90.5	70.1	0.327	0.674	77.1	81.3
	Llama2-13b	78.4	**92.0**	70.8	**0.308**	0.745	81.9	87.3
	Vicuna-7b	88.7	80.7	69.8	0.384	0.388	57.2	79.3
	Vicuna-13b	88.8	81.6	70.8	0.375	0.409	58.4	83.2
	Mistral-7b	83.7	85.5	69.5	0.363	0.520	66.7	96.5
Contriever	**Llama2-7b**	85.3	91.0	**76.6**	0.940	0.674	75.0	79.6
	Llama2-13b	82.0	**92.0**	74.2	0.936	0.740	80.7	87.3
	Vicuna-7b	**92.1**	81.5	73.9	0.948	0.391	55.1	**76.9**
	Vicuna-13b	91.3	83.2	74.7	0.950	**0.376**	**53.5**	79.5
	Mistral-7b	89.2	86.6	75.9	0.942	0.514	63.1	95.3
w/o *GARAG*		-	-	-	1.000	1.000	100	100

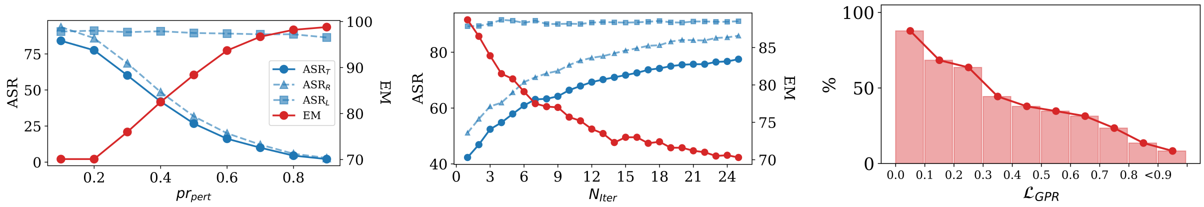

Figure 3: (Left & Center) Adversarial attack results depending on the number of iterations N_{iter}, on NQ with Contriever and Llama2-7b. (Right) Distribution of incorrectness among predictions with the Contriever and Llama-7b depending on $\mathcal{L}_{\text{GPR}}$.

4 Experimental Setup

In this section, we describe the experimental setup.

4.1 Model

Retriever. We use two recent dense retrievers: **DPR** (Karpukhin et al., 2020), a supervised one trained on query-document pairs, and **Contriever** (Izacard et al., 2022), an unsupervised one.
Reader. Following concurrent work (Asai et al., 2024; Wang et al., 2024) that utilizes LLMs as readers for the RAG system, with parameters ranging from 7B to 13B, we have selected open-source LLMs of similar capacities: **Llama2** (Touvron et al., 2023), **Vicuna** (Chiang et al., 2023), and **Mistral** (Jiang et al., 2023). Each model has been either chat-versioned or instruction-tuned. To adapt these models for open-domain QA tasks, we employ a zero-shot prompting template for exact match QA derived from Wang et al. (2024).

4.2 Dataset

We leverage three representative QA datasets: **Natural Questions (NQ)** (Kwiatkowski et al., 2019), **TriviaQA (TQA)** (Joshi et al., 2017), and **SQuAD (SQD)** (Rajpurkar et al., 2016), following the setups of Karpukhin et al. (2020). To assess the robustness of the RAG system, we randomly extract 1,000 instances of the triple $(\boldsymbol{q}, \boldsymbol{d}, \boldsymbol{a})$. In each triple, $\boldsymbol{q}$ is a question from the datasets, $\boldsymbol{d}$ is a document from the top-100 documents retrieved from the Wikipedia corpus corresponding to $\boldsymbol{q}$, and $\boldsymbol{a}$ is the answer generated by the LLM, which is considered as correct for the specific question-document pair.

4.3 Evaluation Metric

Since we aim to measure how the generated adversarial documents with *GARAG* attack the RAG system, we incorporate three types of metrics to show 1) the overall effectiveness of the adversarial attacks, 2) the adversarial impact of the adversarial samples for each retriever and reader component, and 3) the end-to-end QA performance.
Attack Success Ratio (ASR). Attack Success Ratio (ASR) measures the effectiveness of the adversarial document $\boldsymbol{d'}$ in disrupting the RAG system compared to the original document $\boldsymbol{d}$. Specifically, it is quantified by the proportion of adversarial documents located in the holistic error zone by the proportion of adversarial documents that achieve values below 1 in our objective functions. ASR_R and ASR_L denote the ratios of documents meeting such criteria for each objective function $\mathcal{L}_{\text{RSR}}$, $\mathcal{L}_{\text{GPR}}$, respectively, while ASR_T denotes the documents that satisfy them simultaneously.
Component Error (C.E.). To assess the impact of $\boldsymbol{d^*}$ located in the holistic error zone on each component of RAG, we utilize **Retrieval Error (R.E.)** and **Grounding Error (G.E.)**. Specifically, RE

measures the average of $\mathcal{L}_{\text{RSR}}$ values, indicating the relative relevance score compared to the original document. Then, G.E. measures the proportion of predictions that exactly match the actual answers, measuring the grounding capability to noisy documents. Lower values of each metric mean that they are more vulnerable to adversarial documents.

End-to-End Performance (E2E). To assess how *GARAG* influences end-to-end performance, we report it with standard QA metrics: **Exact Match (EM)** and **Accuracy (Acc)**. In cases when the attack fails, we report the scores using the original document $\boldsymbol{d}$ instead of the adversarial one $\boldsymbol{d}'$.

4.4 Implementation Details

The proposed method, *GARAG*, was configured with hyperparameters: N_{iter} was set to 25, N_{parents} to 10, and S to 25. pr_{per}, pr_{cross}, and pr_{mut} were set to 0.2, 0.2, and 0.4, respectively. The operations of perturbation function f in *GARAG* consist of the inner swap, truncate, keyboard typo, and natural typo, following Eger and Benz (2020)[3]. For computing resources, we use A100 GPU clusters.

5 Results

In this section, we show our experimental results with an in-depth analysis of the adversarial attack.

Main Result. Table 1 shows our main results averaged over three datasets using *GARAG* with three metrics: attack success ratio (ASR), components error (C.E.), and end-to-end performance (E2E). First, a notable success rate of over 70% across all scenarios indicates that *GARAG* effectively locates adversarial documents within the holistic error zone by simultaneously considering retrieval and reader errors. This also implies that the RAG system is vulnerable to low-level (yet realistic) perturbations. Additionally, the results indicate that two different retrievers show varying susceptibilities to attacks: Contriever is more vulnerable than DPR. Furthermore, the results reveal that an increase in model size does not necessarily enhance robustness to adversarial attacks, as shown by the minimal differences in ASR between LLMs with 7B and 13B parameters. This suggests that simply increasing the size may not be an optimal solution when addressing the realistic challenges in RAG.

Then, how does an optimal adversarial document located in the holistic error zone specifically influence each component within the RAG system? To answer this, we analyze its impact on both the retrieval and reader components by measuring C.E. Interestingly, the results indicate that adversarial documents within the holistic error zone do not affect the retriever and reader components of different models to the same extent. Note that a higher ASR does not necessarily result in lower C.E. for each component. In detail, although DPR exhibits a significantly lower ASR compared to Contriever, its Retrieval Error (R.E.) remains significantly low, consistently below 0.5. This suggests that adversarial documents targeting DPR are ranked higher in the retrieval corpora, indicating a more effective disruption despite fewer successful attacks. On the other hand, Contriever is more susceptible to attacks, but the impact of these attacks tends to be relatively smaller. Furthermore, although Vicuna appears to be the least vulnerable according to its ASR, it suffers the most significant effects from successful adversarial attacks, as indicated by its Grounding Error (G.E.).

Finally, we further analyze the E2E performance to assess how adversarial attacks impact overall QA performance. Based on the EM metric, the performance of RAG systems decreased by an average of 30% and a maximum of close to 50% in all cases. These findings imply that noisy documents with minor errors, frequently found in real-world databases, can pose significant risks to downstream tasks using RAG. Additionally, we find that the robustness of an RAG system varies significantly depending on the specific retriever and LLMs targeted, thus necessitating the need for careful design of both retrievers and readers to address challenges in robust RAG applications effectively.

Impact of Hyperparameter. We further explore how varying the perturbation probability pr_{pert} and the number of iterations N_{iter} affects the attack outcomes. As the left and center figures of Figure 3 illustrate, there is an apparent correlation between the attack success rates for the retriever (ASR$_R$) and the entire pipeline (ASR$_T$) while also revealing a significant vulnerability in the reader as indicated by the high success rate for the LLM (ASR$_L$). Interestingly, in the left figure of Figure 3, the results indicate that a lower proportion of perturbation within a document leads to a more disruptive impact on the RAG system. This poses a significant concern, given that documents with a few typos are commonly found in the real world. Overall, these findings highlight the critical role of the retriever as a first line of defense in the entire RAG system.

[3] https://github.com/yannikbenz/zeroe

Table 2: Case study with Contriever and Llama-7b, where perturbed texts are in red and correct answers are in blue.

Question	Name a food you might eat on thanksgiving.
Noisy Document	Thanksgivong (8nited States) the Pilgrims who settled at Plymouth Plantation. It is continued in modern times with the Thanksgiving dinner, traditionally featuring turkey, playing a central ro;e in the celebartion of Thanksgiving. In the United States, cetrain kinds of good are traditionally served at Thanksgiving meals. Turkey, usualla roasted and stuffed (but sometimes deep-fried instead), is typically the feat8red!25 item on most Thanksgiving feast tables, so much so that Thanksgiving is also colloquially known as" Turkey Day." In fact, 45 mollion turkeys were consumed on Thanksgiving Day alone in 2015. With 85 percent of Americans partaking in the meal, that's an estimated 276.
Answer	Turkey
Prediction	Mashed potatoes

Table 3: Results of punctuation insertion, phonetic swap, and visual swap on NQ with Contriever and Llama-7b.

	ASR			C.E.		E2E
Attack	ASR_R	ASR_L	ASR_T	R.E.	G.E.	EM
Typo	85.9	91.1	77.5	0.96	63.0	70.1
Punc.	93.0	93.7	86.7	0.91	65.8	68.9
Phonetic.	84.7	92.1	76.8	0.96	62.3	70.0
Visual.	77.7	90.5	68.8	0.98	61.0	72.5

Table 4: Ablation studies of assessing the impact of each step within *GARAG* on NQ with Contriever and Llama-7b.

	ASR			
Attack	ASR_R	ASR_L	ASR_T	N_{iter}
GARAG	**85.9**	**91.1**	**77.5**	**14.8**
w/o Cross. & Mutat.	83.0	90.7	73.7	15.6
w/o Select.	79.4	89.9	69.5	15.6

Impact of Lowering $\mathcal{L}_{\text{GPR}}$. Since the value of $\mathcal{L}_{\text{RSR}}$ does not directly indicate the likelihood of generating incorrect answers with auto-regressive models, we analyze the correlation between the likelihood of generating incorrect answers and $\mathcal{L}_{\text{GPR}}$. As illustrated in the right figure of Figure 3, we categorize predictions into buckets based on their $\mathcal{L}_{\text{GPR}}$ ranges and calculate the proportion of incorrect answers within each bucket. The results indicate that a lower $\mathcal{L}_{\text{GPR}}$ value is correlated with a higher likelihood of incorrect responses, thus corroborating our objective design.

Other Low-level Perturbations. While focusing on character-level perturbations, we also investigate other low-level yet prevalent disturbances, such as punctuation insertion (Formento et al., 2023) and character swaps based on phonetic or visual similarities (Eger et al., 2019; Le et al., 2022). As shown in Table 3, these perturbations show higher success rates and lower E2E performance than those with typos, with punctuation insertion alone compromising the RAG in 86% of attacks. The results emphasize the RAG system's susceptibility to diverse low-level perturbations.

Ablation Study. We conducted ablation studies to see how each step in *GARAG* contributes to the overall performance. As shown in Table 4, omitting the crossover and mutation steps results in a lower ASR, reducing the attack's overall effectiveness due to limited search space exploration. Furthermore, without the selection step, lower ASR_R indicates that the optimization becomes unbalanced. Overall, each step in *GARAG* plays a crucial role in achieving a balanced optimization during attacks targeting both the retriever and reader components.

Case Study. We further qualitatively assess the impact of low-level textual perturbations within a document in Table 2. Note that since we ensure that the answer spans remain unperturbed, the readers should ideally generate correct answers. However, interestingly, an LLM fails to identify the correct answer, "Turkey", which is mentioned four times in the document, but instead generates "Mashed potatoes", which is never mentioned at all. We include more diverse cases in Table 6.

6 Conclusion

In this work, we highlighted the importance of assessing the overall robustness of the retriever and reader components within the RAG system, particularly against noisy documents containing minor typos that are common in real-world databases. Specifically, we proposed two objectives to evaluate the resilience of each component, focusing on their sequential dependencies. Furthermore, to simulate real-world noises with low-level perturbations, we introduced a novel adversarial attack method, *GARAG*, incorporating a genetic algorithm. Our findings indicate that noisy documents critically hurt the RAG system, significantly degrading its performance. Although the retriever serves as a protective barrier for the reader, it still remains susceptible to minor disruptions. Our *GARAG* shows promise as an adversarial attack strategy when assessing the holistic robustness of RAG systems against various low-level perturbations.

Limitation

In this work, we explored the robustness of the RAG system by using various recent open-source LLMs of different sizes, which are widely used as reader components in this system. However, due to our limited academic budget, we could not include much larger black-box LLMs such as the GPT series models, which have a hundred billion parameters. We believe that exploring the robustness of these LLMs as reader components would be a valuable line of future work. Furthermore, *GARAG* aims for the optimal adversarial document to be located within a holistic error zone, by simultaneously considering both retrieval and grounding errors. However, we would like to note that even though the adversarial document is located within the holistic error zone, this does not necessarily mean that the reader will always generate incorrect answers for every query, due to the auto-regressive nature of how reader models generate tokens. Nevertheless, as shown in the right figure of Figure 3 and discussed in its analysis, we would like to emphasize that there is a clear correlation: a lower $\mathcal{L}_{\text{GPR}}$ value is associated with a higher likelihood of incorrect responses.

Ethics Statement

We designed a novel attack strategy for the purpose of building robust and safe RAG systems when deployed in the real world. However, given the potential for malicious users to exploit our *GARAG* and deliberately attack the system, it is crucial to consider these scenarios. Therefore, to prevent such incidents, we also present a defense strategy, detailed in Figure 4 and its analysis. Additionally, we believe that developing a range of defense strategies remains a critical area for future work.

References

Moustafa Alzantot, Yash Sharma, Ahmed Elgohary, Bo-Jhang Ho, Mani B. Srivastava, and Kai-Wei Chang. 2018. Generating natural language adversarial examples. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing, Brussels, Belgium, October 31 - November 4, 2018*, pages 2890–2896. Association for Computational Linguistics.

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*.

Jinheon Baek, Soyeong Jeong, Minki Kang, Jong C. Park, and Sung Ju Hwang. 2023. Knowledge-augmented language model verification. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing, EMNLP 2023, Singapore, December 6-10, 2023*, pages 1720–1736. Association for Computational Linguistics.

Tom B. Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel M. Ziegler, Jeffrey Wu, Clemens Winter, Christopher Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. 2020. Language models are few-shot learners. In *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual*.

Harrison Chase. 2022. LangChain.

Jiawei Chen, Hongyu Lin, Xianpei Han, and Le Sun. 2024. Benchmarking large language models in retrieval-augmented generation. In *Thirty-Eighth AAAI Conference on Artificial Intelligence, AAAI 2024, Thirty-Sixth Conference on Innovative Applications of Artificial Intelligence, IAAI 2024, Fourteenth Symposium on Educational Advances in Artificial Intelligence, EAAI 2014, February 20-27, 2024, Vancouver, Canada*, pages 17754–17762. AAAI Press.

Wei-Lin Chiang, Zhuohan Li, Zi Lin, Ying Sheng, Zhanghao Wu, Hao Zhang, Lianmin Zheng, Siyuan Zhuang, Yonghao Zhuang, Joseph E. Gonzalez, Ion Stoica, and Eric P. Xing. 2023. Vicuna: An open-source chatbot impressing gpt-4 with 90%* chatgpt quality.

Sukmin Cho, Jeongyeon Seo, Soyeong Jeong, and Jong C. Park. 2023. Improving zero-shot reader by reducing distractions from irrelevant documents in open-domain question answering. In *Findings of the Association for Computational Linguistics: EMNLP 2023, Singapore, December 6-10, 2023*, pages 3145–3157. Association for Computational Linguistics.

Florin Cuconasu, Giovanni Trappolini, Federico Siciliano, Simone Filice, Cesare Campagnano, Yoelle Maarek, Nicola Tonellotto, and Fabrizio Silvestri. 2024. The power of noise: Redefining retrieval for RAG systems. *arXiv preprint arXiv:2401.14887*, abs/2401.14887.

Kalyanmoy Deb, Samir Agrawal, Amrit Pratap, and T. Meyarivan. 2002. A fast and elitist multiobjective genetic algorithm: NSGA-II. *IEEE Trans. Evol. Comput.*, 6(2):182–197.

Mohammad Dehghan, Dhruv Kumar, and Lukasz Golab. 2022. GRS: combining generation and revision in unsupervised sentence simplification. In *Findings of the Association for Computational Linguistics: ACL 2022, Dublin, Ireland, May 22-27, 2022*, pages 949–960. Association for Computational Linguistics.

Javid Ebrahimi, Anyi Rao, Daniel Lowd, and Dejing Dou. 2018. Hotflip: White-box adversarial examples for text classification. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics, ACL 2018, Melbourne, Australia, July 15-20, 2018, Volume 2: Short Papers*, pages 31–36. Association for Computational Linguistics.

Steffen Eger and Yannik Benz. 2020. From hero to zéroe: A benchmark of low-level adversarial attacks. In *Proceedings of the 1st Conference of the Asia-Pacific Chapter of the Association for Computational Linguistics and the 10th International Joint Conference on Natural Language Processing, AACL/IJCNLP 2020, Suzhou, China, December 4-7, 2020*, pages 786–803. Association for Computational Linguistics.

Steffen Eger, Gözde Gül Sahin, Andreas Rücklé, Ji-Ung Lee, Claudia Schulz, Mohsen Mesgar, Krishnkant Swarnkar, Edwin Simpson, and Iryna Gurevych. 2019. Text processing like humans do: Visually attacking and shielding NLP systems. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2019, Minneapolis, MN, USA, June 2-7, 2019, Volume 1 (Long and Short Papers)*, pages 1634–1647. Association for Computational Linguistics.

Manaal Faruqui, Ellie Pavlick, Ian Tenney, and Dipanjan Das. 2018. Wikiatomicedits: A multilingual corpus of wikipedia edits for modeling language and discourse. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing, Brussels, Belgium, October 31 - November 4, 2018*, pages 305–315. Association for Computational Linguistics.

Brian Formento, Chuan-Sheng Foo, Anh Tuan Luu, and See-Kiong Ng. 2023. Using punctuation as an adversarial attack on deep learning-based NLP systems: An empirical study. In *Findings of the Association for Computational Linguistics: EACL 2023, Dubrovnik, Croatia, May 2-6, 2023*, pages 1–34. Association for Computational Linguistics.

Gautier Izacard, Mathilde Caron, Lucas Hosseini, Sebastian Riedel, Piotr Bojanowski, Armand Joulin, and Edouard Grave. 2022. Unsupervised dense information retrieval with contrastive learning. *Trans. Mach. Learn. Res.*, 2022.

Soyeong Jeong, Jinheon Baek, Sukmin Cho, Sung Ju Hwang, and Jong C. Park. 2024. Adaptive-RAG: Learning to adapt retrieval-augmented large language models through question complexity. In *2024 Annual Conference of the North American Chapter of the Association for Computational Linguistics*.

Albert Q. Jiang, Alexandre Sablayrolles, Arthur Mensch, Chris Bamford, Devendra Singh Chaplot, Diego de Las Casas, Florian Bressand, Gianna Lengyel, Guillaume Lample, Lucile Saulnier, Lélio Renard Lavaud, Marie-Anne Lachaux, Pierre Stock, Teven Le Scao, Thibaut Lavril, Thomas Wang, Timothée Lacroix, and William El Sayed. 2023. Mistral 7b. *arXiv preprint arXiv:2310.06825*, abs/2310.06825.

Di Jin, Zhijing Jin, Joey Tianyi Zhou, and Peter Szolovits. 2020. Is BERT really robust? A strong baseline for natural language attack on text classification and entailment. In *The Thirty-Fourth AAAI Conference on Artificial Intelligence, AAAI 2020, The Thirty-Second Innovative Applications of Artificial Intelligence Conference, IAAI 2020, The Tenth AAAI Symposium on Educational Advances in Artificial Intelligence, EAAI 2020, New York, NY, USA, February 7-12, 2020*, pages 8018–8025. AAAI Press.

Mandar Joshi, Eunsol Choi, Daniel S. Weld, and Luke Zettlemoyer. 2017. Triviaqa: A large scale distantly supervised challenge dataset for reading comprehension. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics, ACL 2017, Vancouver, Canada, July 30 - August 4, Volume 1: Long Papers*, pages 1601–1611. Association for Computational Linguistics.

Vladimir Karpukhin, Barlas Oguz, Sewon Min, Patrick S. H. 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 2020, Online, November 16-20, 2020*, pages 6769–6781. Association for Computational Linguistics.

Jungo Kasai, Keisuke Sakaguchi, Yoichi Takahashi, Ronan Le Bras, Akari Asai, Xinyan Yu, Dragomir Radev, Noah A. Smith, Yejin Choi, and Kentaro Inui. 2023. Realtime QA: what's the answer right now? In *Advances in Neural Information Processing Systems 36: Annual Conference on Neural Information Processing Systems 2023, NeurIPS 2023, New Orleans, LA, USA, December 10 - 16, 2023*.

Tom Kwiatkowski, Jennimaria Palomaki, Olivia Redfield, Michael Collins, Ankur P. Parikh, Chris Alberti, Danielle Epstein, Illia Polosukhin, Jacob Devlin, Kenton Lee, Kristina Toutanova, Llion Jones, Matthew Kelcey, Ming-Wei Chang, Andrew M. Dai, Jakob Uszkoreit, Quoc Le, and Slav Petrov. 2019. Natural questions: a benchmark for question answering research. *Trans. Assoc. Comput. Linguistics*, 7:452–466.

Angeliki Lazaridou, Elena Gribovskaya, Wojciech Stokowiec, and Nikolai Grigorev. 2022. Internet-augmented language models through few-shot prompting for open-domain question answering. *arXiv preprint arXiv:2203.05115*, abs/2203.05115.

Thai Le, Jooyoung Lee, Kevin Yen, Yifan Hu, and Dongwon Lee. 2022. Perturbations in the wild: Leveraging human-written text perturbations for realistic adversarial attack and defense. In *Findings of the Association for Computational Linguistics: ACL 2022, Dublin, Ireland, May 22-27, 2022*, pages 2953–2965. Association for Computational Linguistics.

Thai Le, Yiran Ye, Yifan Hu, and Dongwon Lee. 2023. Cryptext: Database and interactive toolkit of human-written text perturbations in the wild. In *39th IEEE International Conference on Data Engineering, ICDE 2023, Anaheim, CA, USA, April 3-7, 2023*, pages 3639–3642. IEEE.

Hyunji Lee, Se June Joo, Chaeeun Kim, Joel Jang, Doyoung Kim, Kyoung-Woon On, and Minjoon Seo. 2023. How well do large language models truly ground? *arXiv preprint arXiv:2311.09069*, abs/2311.09069.

Patrick S. H. Lewis, Ethan Perez, Aleksandra Piktus, Fabio Petroni, Vladimir Karpukhin, Naman Goyal, Heinrich Küttler, Mike Lewis, Wen-tau Yih, Tim Rocktäschel, Sebastian Riedel, and Douwe Kiela. 2020. Retrieval-augmented generation for knowledge-intensive NLP tasks. In *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual*.

Junyi Li, Xiaoxue Cheng, Xin Zhao, Jian-Yun Nie, and Ji-Rong Wen. 2023a. Halueval: A large-scale hallucination evaluation benchmark for large language models. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing, EMNLP 2023, Singapore, December 6-10, 2023*, pages 6449–6464. Association for Computational Linguistics.

Xinzhe Li, Ming Liu, Shang Gao, and Wray L. Buntine. 2023b. A survey on out-of-distribution evaluation of neural NLP models. In *Proceedings of the Thirty-Second International Joint Conference on Artificial Intelligence, IJCAI 2023, 19th-25th August 2023, Macao, SAR, China*, pages 6683–6691. ijcai.org.

Jerry Liu. 2022. LlamaIndex.

Quanyu Long, Yue Deng, Leilei Gan, Wenya Wang, and Sinno Jialin Pan. 2024. Backdoor attacks on dense passage retrievers for disseminating misinformation. *arXiv preprint arXiv:2402.13532*, abs/2402.13532.

Alex Mallen, Akari Asai, Victor Zhong, Rajarshi Das, Daniel Khashabi, and Hannaneh Hajishirzi. 2023. When not to trust language models: Investigating effectiveness of parametric and non-parametric memories. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), ACL 2023, Toronto, Canada, July 9-14, 2023*, pages 9802–9822. Association for Computational Linguistics.

OpenAI. 2023a. Chatgpt plugins.

OpenAI. 2023b. GPT-4 technical report. *arXiv preprint arXiv:2303.08774*, abs/2303.08774.

Nicolas Papernot, Patrick D. McDaniel, Ananthram Swami, and Richard E. Harang. 2016. Crafting adversarial input sequences for recurrent neural networks. In *2016 IEEE Military Communications Conference, MILCOM 2016, Baltimore, MD, USA, November 1-3, 2016*, pages 49–54. IEEE.

Aleksandra Piktus, Fabio Petroni, Vladimir Karpukhin, Dmytro Okhonko, Samuel Broscheit, Gautier Izacard, Patrick S. H. Lewis, Barlas Oguz, Edouard Grave, Wen-tau Yih, and Sebastian Riedel. 2021. The web is your oyster - knowledge-intensive NLP against a very large web corpus. *arXiv preprint arXiv:2112.09924*, abs/2112.09924.

Ofir Press, Muru Zhang, Sewon Min, Ludwig Schmidt, Noah A. Smith, and Mike Lewis. 2023. Measuring and narrowing the compositionality gap in language models. In *Findings of the Association for Computational Linguistics: EMNLP 2023, Singapore, December 6-10, 2023*, pages 5687–5711. Association for Computational Linguistics.

Yujia Qin, Shihao Liang, Yining Ye, Kunlun Zhu, Lan Yan, Yaxi Lu, Yankai Lin, Xin Cong, Xiangru Tang, Bill Qian, Sihan Zhao, Lauren Hong, Runchu Tian, Ruobing Xie, Jie Zhou, Mark Gerstein, dahai li, Zhiyuan Liu, and Maosong Sun. 2024. ToolLLM: Facilitating large language models to master 16000+ real-world APIs. In *The Twelfth International Conference on Learning Representations*.

Pranav Rajpurkar, Jian Zhang, Konstantin Lopyrev, and Percy Liang. 2016. Squad: 100, 000+ questions for machine comprehension of text. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing, EMNLP 2016, Austin, Texas, USA, November 1-4, 2016*, pages 2383–2392. The Association for Computational Linguistics.

Shuhuai Ren, Yihe Deng, Kun He, and Wanxiang Che. 2019. Generating natural language adversarial examples through probability weighted word saliency. In *Proceedings of the 57th Conference of the Association for Computational Linguistics, ACL 2019, Florence, Italy, July 28- August 2, 2019, Volume 1: Long Papers*, pages 1085–1097. Association for Computational Linguistics.

Marco Túlio Ribeiro, Sameer Singh, and Carlos Guestrin. 2018. Semantically equivalent adversarial rules for debugging NLP models. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics, ACL 2018, Melbourne, Australia, July 15-20, 2018, Volume 1: Long Papers*, pages 856–865. Association for Computational Linguistics.

Nandan Thakur, Luiz Bonifacio, Xinyu Zhang, Odunayo Ogundepo, Ehsan Kamalloo, David Alfonso-Hermelo, Xiaoguang Li, Qun Liu, Boxing Chen, Mehdi Rezagholizadeh, and Jimmy Lin. 2023. Nomiracl: Knowing when you don't know for robust multilingual retrieval-augmented generation. *arXiv preprint arXiv:2312.11361*, abs/2312.11361.

Nandan Thakur, Nils Reimers, Andreas Rücklé, Abhishek Srivastava, and Iryna Gurevych. 2021. BEIR: A heterogeneous benchmark for zero-shot evaluation of information retrieval models. In *Proceedings of the Neural Information Processing Systems Track on Datasets and Benchmarks 1, NeurIPS Datasets and Benchmarks 2021, December 2021, virtual*.

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, Aurélien Rodriguez, Robert Stojnic, Sergey Edunov, and Thomas Scialom. 2023. Llama 2: Open foundation and fine-tuned chat models. *arXiv preprint arXiv:2307.09288*, abs/2307.09288.

Boxin Wang, Weixin Chen, Hengzhi Pei, Chulin Xie, Mintong Kang, Chenhui Zhang, Chejian Xu, Zidi Xiong, Ritik Dutta, Rylan Schaeffer, Sang T. Truong, Simran Arora, Mantas Mazeika, Dan Hendrycks, Zinan Lin, Yu Cheng, Sanmi Koyejo, Dawn Song, and Bo Li. 2023. Decodingtrust: A comprehensive assessment of trustworthiness in GPT models. In *Advances in Neural Information Processing Systems 36: Annual Conference on Neural Information Processing Systems 2023, NeurIPS 2023, New Orleans, LA, USA, December 10 - 16, 2023*.

Boxin Wang, Chejian Xu, Shuohang Wang, Zhe Gan, Yu Cheng, Jianfeng Gao, Ahmed Hassan Awadallah, and Bo Li. 2021. Adversarial GLUE: A multi-task benchmark for robustness evaluation of language models. In *Proceedings of the Neural Information Processing Systems Track on Datasets and Benchmarks 1, NeurIPS Datasets and Benchmarks 2021, December 2021, virtual*.

Yuhao Wang, Ruiyang Ren, Junyi Li, Wayne Xin Zhao, Jing Liu, and Ji-Rong Wen. 2024. REAR: A relevance-aware retrieval-augmented framework for open-domain question answering. *arXiv preprint arXiv:2402.17497*, abs/2402.17497.

Phoenix Neale Williams and Ke Li. 2023. Black-box sparse adversarial attack via multi-objective optimisation CVPR proceedings. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2023, Vancouver, BC, Canada, June 17-24, 2023*, pages 12291–12301. IEEE.

Jin Yong Yoo and Yanjun Qi. 2021. Towards improving adversarial training of NLP models. In *Findings of the Association for Computational Linguistics: EMNLP 2021, Virtual Event / Punta Cana, Dominican Republic, 16-20 November, 2021*, pages 945–956. Association for Computational Linguistics.

Ori Yoran, Tomer Wolfson, Ori Ram, and Jonathan Berant. 2024. Making retrieval-augmented language models robust to irrelevant context. In *The Twelfth International Conference on Learning Representations*.

Yuan Zang, Fanchao Qi, Chenghao Yang, Zhiyuan Liu, Meng Zhang, Qun Liu, and Maosong Sun. 2020. Word-level textual adversarial attacking as combinatorial optimization. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, ACL 2020, Online, July 5-10, 2020*, pages 6066–6080. Association for Computational Linguistics.

Wei Emma Zhang, Quan Z. Sheng, Ahoud Alhazmi, and Chenliang Li. 2020. Adversarial attacks on deep-learning models in natural language processing: A survey. *ACM Trans. Intell. Syst. Technol.*, 11(3):24:1–24:41.

Zexuan Zhong, Ziqing Huang, Alexander Wettig, and Danqi Chen. 2023. Poisoning retrieval corpora by injecting adversarial passages. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing, EMNLP 2023, Singapore, December 6-10, 2023*, pages 13764–13775. Association for Computational Linguistics.

Kaijie Zhu, Jindong Wang, Jiaheng Zhou, Zichen Wang, Hao Chen, Yidong Wang, Linyi Yang, Wei Ye, Neil Zhenqiang Gong, Yue Zhang, and Xing Xie. 2023. Promptbench: Towards evaluating the robustness of large language models on adversarial prompts. *arXiv preprint arXiv:2306.04528*, abs/2306.04528.

Wei Zou, Runpeng Geng, Binghui Wang, and Jinyuan Jia. 2024. Poisonedrag: Knowledge poisoning attacks to retrieval-augmented generation of large language models. *arXiv preprint arXiv:2402.07867*, abs/2402.07867.

A Implementation Detail

A.1 Operations

The operations of transformation function f in our work are as follows:

- **Inner-Shuffle**: Randomly shuffles the letters within a subsequence of a word token, limited to words with more than three characters.
- **Truncate**: Removes a random number of letters from either the beginning or end of a word token. This operation is restricted to words with more than three characters, with a maximum of three characters removed.
- **Keyboard Typo**: Substitutes a letter with its adjacent counterpart on an English keyboard layout to simulate human typing errors. Only one character per word is replaced.
- **Natural Typo**: Replaces letters based on common human errors derived from Wikipedia's edit history. This operation encompasses a variety of error types, including phonetic errors, omissions, morphological errors, and their combinations.

Also, we explore other types of low-level perturbations, such as punctuation insertion and phonetic and visual similarity. The operations of these low-level perturbations are as follows:

- **Punctuation Insertion**: Insert random punctations into the beginning or end of a word token. We insert a maximum of three identical punctuation into the beginning or end of the word. Exploited punctuation is " ,.'!?; ".
- **Phonetic Similarity**: Swap the characters in a word into the other tokens having phonetic similarity with the original ones. We exploit two types of phonetic similarity attacks from Eger and Benz (2020) and Le et al. (2022).
- **Visual Similarity**: Swap the characters in a word into the other tokens having visual similarity with the original ones. We exploit two types of phonetic similarity attacks from Eger et al. (2019).

Algorithm 1: Genetic Attack on RAG

Input: Query $\boldsymbol{q}$, Document $\boldsymbol{d}$, Number of iterations N_{iter}, Number of parents N_{parent}, Population size S, Perturbation rate pr_{per}, Crossover rate pr_{cross}, Mutation rate pr_{mut}
Function: Non-dominated sorting NDS, Crowd sorting CS
Output: Adversarial document $\boldsymbol{d'}^*$

```
// Initialization
P_0 ← {d'_i}_{i=1}^S with pr_per;
for i = 1 to N_iter do
    // Crossover
    O ← CROSSOVER(P_{i-1}, N_parent, pr_cross);
    // Mutation
    O ← MUTATE(O, pr_mut);
    // Selection
    P̂_i ← P_{i-1} ∪ O;
    for d' in P̂_i do
        Evaluate L_RSR(d') and L_GPR(d');
    P̂_i ← CS(NDS(P̂_i));
    d* ← Top-1(P̂_i) ;
    if a ≠ LLM(d*, q; θ) and L_RSR(d*) < 1 then
        return d* as adversarial example;
    P_i ← Top-S(P̂_i);
d* ← Top-1(P_{N_iter}) ;
return d* as adversarial example;
```

A.2 Process of *GARAG*

The detailed process of *GARAG* is showcased in Algorithm 1. Our process begins with the initialization of the adversarial document population, and then the population repeats the cycles of crossover, mutation, and selection.

A.3 Template

We adopt the zero-shot prompting template optimal for exact QA tasks, derieved from Wang et al. (2024), for all LLMs exploited in our experiments.

QA Template for LLMs

[INST] Documents:
{Document}

Answer the following question with a very short phrase, such as "1998", "May 16th, 1931", or "James Bond", to meet the criteria of exact match datasets.

Question: {Question} [/INST]

Answer:

Table 5: Adversarial attack results of *GARAG* on three QA datasets across different retrievers and LLMs.

		NQ							TriviaQA							SQuAD						
		ASR(↑)			**C.E.(↓)**		**E2E(↓)**		**ASR(↑)**			**C.E.(↓)**		**E2E(↓)**		**ASR(↑)**			**C.E.(↓)**		**E2E(↓)**	
Retriever	**LLM**	ASR_R	ASR_L	ASR_T	RE.	GE.	EM	Acc.	ASR_R	ASR_L	ASR_T	RE.	GE.	EM	Acc.	ASR_R	ASR_L	ASR_T	RE.	GE.	EM	Acc.
DPR	**Llama2-7b**	75.4	89.8	66.0	0.387	0.689	76.8	80.6	78.2	91.7	70.2	0.312	0.730	81.6	85.3	84.1	90.1	74.2	0.280	0.637	73.0	78.
	Llama2-13b	71.3	91.7	63.5	0.357	0.695	82.8	88.2	83.9	92.0	76.1	0.266	0.630	76.7	83.3	80.0	92.4	72.7	0.299	0.722	86.3	90.5
	Vicuna-7b	83.0	81.6	65.1	0.423	0.786	62.0	79.2	91.1	79.5	70.8	0.391	0.775	58.4	81.7	92.0	81.1	73.4	0.338	0.742	51.2	76.9
	Vicuna-13b	82.8	80.9	64.4	0.423	0.77	58.5	83.3	91.8	83.5	75.4	0.367	0.779	59.2	85.7	91.7	80.5	72.5	0.336	0.722	57.4	80.5
	Mistral-7b	78.5	85.9	65.1	0.397	0.8	69.1	96.5	84.7	84.9	69.8	0.352	0.811	66.5	97.7	87.8	85.7	73.5	0.34	0.701	64.4	95.2
Contriever	**Llama2-7b**	85.9	91.1	77.5	0.941	0.639	70.1	74.7	84.9	90.7	76.0	0.94	0.725	82.0	86.9	85.2	91.2	76.4	0.94	0.605	72.9	77.2
	Llama2-13b	78.9	91.2	70.5	0.939	0.647	78.7	85.7	81.0	91.9	72.9	0.932	0.723	86.2	91.7	86.1	93.0	79.1	0.938	0.633	77.2	84.5
	Vicuna-7b	90.8	81.3	72.4	0.949	0.738	52.2	72.5	93.0	80.8	74.0	0.946	0.764	60.3	81.5	92.6	82.5	75.2	0.948	0.712	52.7	76.7
	Vicuna-13b	87.5	85.5	73.3	0.94	0.735	63.9	95.4	88.8	86.4	75.2	0.944	0.796	66.2	97.8	91.2	88.0	79.3	0.942	0.704	59.2	92.6
	Mistral-7b	87.5	85.5	73.3	0.94	0.735	63.9	95.4	88.8	86.4	75.2	0.944	0.796	66.2	97.8	91.2	88.0	79.3	0.942	0.704	59.2	92.6

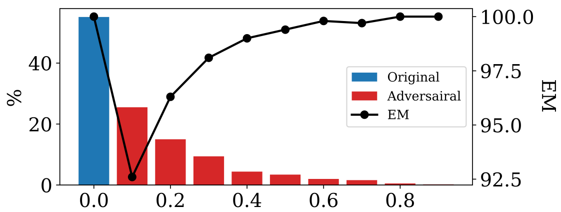

Figure 4: Distribution of grammatically correct document among $\boldsymbol{d}^*$ on NQ with the Contriever and Llama2-7b.

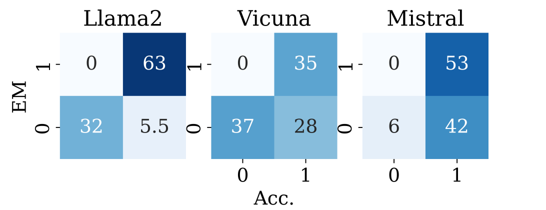

Figure 5: Correlation matrices of prediction from the adversarial document $\boldsymbol{d}^*$ across EM and Acc. with Contriever.

B Additional Results

B.1 Overall Result

Table 5 shows the overall results across three QA datasets, two retrievers, and five LLMs.

B.2 Defense Strategy.

Various defense mechanisms against adversarial attacks in NLP have been proposed. Adversarial training, fine-tuning the model on adversarial samples, is a popular approach (Yoo and Qi, 2021). However, this strategy is not practically viable for RAG systems, given the prohibitive training costs associated with models exceeding a billion parameters. Alternatively, a grammar checker is an effective defense against low-level perturbations within documents (Formento et al., 2023).

Our analysis, depicted in Figure 4, compares the grammatical correctness of original and adversarial documents via grammar checker model [4] presented in Dehghan et al. (2022). It reveals that approximately 50% of the original samples contain grammatical errors. Also, even within the adversarial set, about 25% of the samples maintain grammatical correctness at a low perturbation level. This observation highlights a critical limitation: relying solely on a grammar checker would result in dismissing many original documents and accepting some adversarial ones. Consequently, this underscores the limitations of grammar checkers as a standalone defense and points to more sophisticated and tailored defense strategies.

B.3 Analysis on Prediction

We analyze the discrepancy between them with the responding patterns of diverse LLMs when affected by adversarial documents, categorizing results based on EM and Acc values in Figure 5. Specifically, EM strictly assesses whether the prediction exactly matches the correct answer, while Acc assesses only whether the answer span is included within the predicted response. When EM is 0 and Acc is 1 (i.e., (0,1)), the answer span is included along with extraneous tokens. By contrast, when EM is 0 and Acc is 0 (i.e., (0,0)), the answer span is entirely incorrect, indicating a hallucinated prediction. Therefore, Llama2 demonstrates a higher tendency to generate responses that exactly match the annotated samples, as indicated by the high portion of (1,1). However, given its lower proportion of (1,0) results, it frequently produces entirely incorrect answers when exposed to adversarial conditions. By contrast, Mistral, while generating fewer exact matches compared to Llama2, more consistently includes the correct answer span in its responses. These findings are crucial for understanding the behavior of different models in real-world scenarios, particularly in how they handle documents containing noise or adversarial modifications. Therefore, these results show that the

[4] https://huggingface.co/imohammad12/GRS-Grammar-Checker-DeBerta

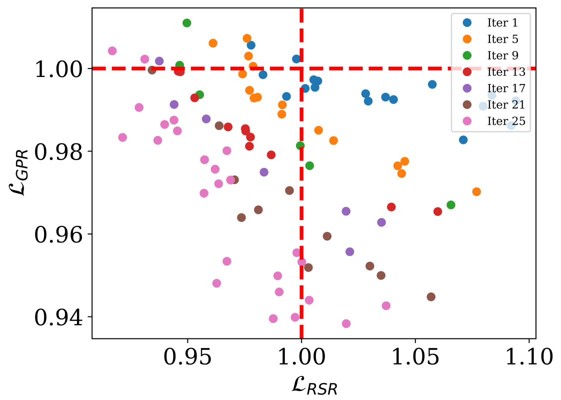

Figure 6: The process of population refinement by *GARAG* on NQ with Contriever and Llama-7b

patterns of LLMs are varied under the influence of noisy documents.

B.4 Case Study

We conducted case studies with diverse LLMs, including Llama-7b, Vicuna-7b, and Mistral-7b, as shown in Table 6. In all these studies, while the answer tokens were not perturbed—allowing for the possibility of grounding correct information—the LLMs typically failed to locate the correct knowledge within the document. This often resulted in incorrect predictions or even hallucinations, where the answer was not just wrong but absent from the document. However, there was an exception with Mistral-7b, which generated the correct answer and additional explanatory text. While this prediction did not meet the Exact Match (EM) metric, it was semantically correct. Additionally, we provide a detailed overview of how the population is refined through the iterative process, as illustrated in Figure 6.

Table 6: Case study on NQ with Contriever, where perturbed texts are in red and correct answers are in blue.

	Llama-7b
Question	Which site of an enzyme is called allosteric site?
Noisy Document	A;losteric enzyme Long-range allostery is esprcially imponant in cell signaling. Allosteric regulation is also particularly important in the cell's abil9ty to adjusy enzyme activity. The term "allostery" comes from the Greek "allos" (), "other," and "stereos" (), " silid (object)." This is in reference to the fact that the regulatory site of an allosteric protein is physically distinct from its active site. The protein catalyst (enzyme) may be paft of a multi-subunit complex, and/or may transiently or permanently associatr with a Cofactor (e.g. adenosine triphosphate). Catalysis of biochejical reactions is vital due to the very law reaction rates of the uncatalysed reactioms .
Answer	Regulatory site
Prediction	Active site
Question	Who did Cora marry in once upon a time?
Noisy Document	The Millerś Daughter (Once Upon a Time) to the King and accepts Henry 's resultant marriage proposal. The day before her wedding, Cora ques6ions her olans ; she is unlikely to become qjeen as Henry is fifth in linf to the throne, while Rumplestiltskin, with whom she has been having an affair, offers her love. They agree to amend the contratc so Cora owes Rumplestiltskin "his" child. He also agrees to teach her how to take a heart, so that she can kilk King Savier. That night, she confromts the king. He reveals that he knows of her relationship with Rumplestiltskin; telling her that " pove is weakness," he ays
Answer	Henry
Prediction	Rumplestiltskin
	Vicuna-7b
Question	What is the 3rd largest state in USA?
Noisy Document	Wextern United States LGBT community, and Oakland, California has a large percen5age of residents being African-American, as well as Long Beadh , California which also has a large Black community. Ths state of Utah has a Mormon majority (estimate at 62.4% in 2004), while some cities like Albuquerque, Nrw Mexico; Billkngs , Montqna ; Spokane, Washington; and Tucson, Arizona are located near Indian Reservations. In remote areas there are settlements of Alaskan Natives and Native Hawaiians. Fqcing both the Pacitic Ocean and the Mexican border, the West has been shaped by a cariety of ethnic groups. Hawaii is the only state in the union in which
Answer	California, Alaska
Prediction	Oregon
Question	When did the movie peter pan come out
Noisy Document	Peter Pan as an adult by Robin Williams, with iblue eyes and dark brown hair; in flashbacks to him in his youth, his hwir is light brown. In this film his ears ap;ear poin only when he is Petef Pan, not as Peter Banning. His Pan attire resembles the D9sney outfit (minus the cap). In the lvie-action 2003 " -eter Pan" film, he is portrayed by Jeremy Su,pter , who has blond hair and blue-green eyes. His outfit is made of leaves and vines. J.M. Barrie created his character based on his older brother, Davic , who died in an ice-skating accieent the day before
Answer	25 December 2003, 2003
Prediction	1998
	Mistral-7b
Question	Make it or break it who goes to the Olympics?
Noisy Document	Make It or Break It A new gymnast, Max (Josh Bowman), cpmes to The Rock, wttracting the attention of Lauren and Payson . Though Max seems more interested in Payson , she is more focused on her dream. Lauren tells Payson that Max is her nww boyfriend; Max denies this. Mas, Austin, and Payson hope Kaylie train in secret, as they try to convince Sasha that Kaylie is ready to compete at Worlds. In the week of Emily's departure, Kelly Parke becomes closer to Kaylie , des-ite the scheming of her stage her, Sheila (Kathy Najimy). Kaylie offers her comfort as Kelly's mom is only sup[ortive of Kelly's
Answer	Jordan, Payson, Lauren, Colleen, Kaylie
Prediction	Kaylie goes to the Olympics
Question	When did Michael Jordan get his last ring?
Noisy Document	Hisfory of basketball A large place of this is due to the transcendent stars that have played the game through the years. It was because of the play of Michael Jordan that basketball started to rsach international audiences, especially on the 992 United States men's Olympic basketball team, known as the Dream Team. After his fianl championship and second retirement in 1998 , there was a void as in who would be the face of basketball. So;n after with the help of Shaquille O'Neal, Koge Bryant would go on to win three straight championships from 2000–2002 with the Los Angeles kers , helping make basketball
Answer	1998
Prediction	2002. (This is based on the context of the given document stating that Kobe Bryant and Shaquille O'Neal)