Deceiving Question-Answering Models: A Hybrid Word-Level Adversarial Approach

Jiyao Li[1], Mingze Ni[1], Yongshun Gong[2], Wei Liu[1*]

[1]School of Computer Science, University of Technology Sydney, 15 Broadway, Sydney, 2007, NSW, Australia.
[2]School of Software, Shandong University, 27 Shanda Nanlu, Jinan, 250100, Shandong, China.

*Corresponding author(s). E-mail(s): wei.liu@uts.edu.au;
Contributing authors: jiyao.li-1@student.uts.edu.au;
mingze.ni@uts.edu.au; ysgong@sdu.edu.cn;

Abstract

Deep learning underpins most of the currently advanced natural language processing (NLP) tasks such as textual classification, neural machine translation (NMT), abstractive summarization and question-answering (QA). However, the robustness of the models, particularly QA models, against adversarial attacks is a critical concern that remains insufficiently explored. This paper introduces QA-Attack (Question Answering Attack), a novel word-level adversarial strategy that fools QA models. Our attention-based attack exploits the customized attention mechanism and deletion ranking strategy to identify and target specific words within contextual passages. It creates deceptive inputs by carefully choosing and substituting synonyms, preserving grammatical integrity while misleading the model to produce incorrect responses. Our approach demonstrates versatility across various question types, particularly when dealing with extensive long textual inputs. Extensive experiments on multiple benchmark datasets demonstrate that QA-Attack successfully deceives baseline QA models and surpasses existing adversarial techniques regarding success rate, semantics changes, BLEU score, fluency and grammar error rate.

1

1 Introduction

Question-answering (QA) models, a key task within Sequence-to-Sequence (Seq2Seq) frameworks, aim to enhance computers' ability to process and respond to natural language queries. As these models have evolved, they have been widely adopted in real-world applications such as customer service chatbots[40], search engines [69], and information retrieval in fields like medicine [19] and law [35]. However, despite the significant progress in deep learning and natural language processing (NLP), these models remain vulnerable to adversarial examples, leading to misinformation, privacy breaches, and flawed decision-making in critical areas [23, 8, 15, 52]. This highlights the importance of understanding how adversarial examples are generated from the attackers' perspective and potential defense mechanisms — an area that remains under-explored.

QA models are expected to comprehend given texts and questions, providing accurate and contextually relevant answers [50]. These models primarily address two types of questions: Informative Queries and Boolean Queries. The Informative Queries typically begin with interrogative words such as "who," "what," "where," "when," "why," or "how," requiring detailed and specific information from the provided context. Although models like T5 [43], LongT5 [14], and BART [27], which follow an encoder-decoder structure, have demonstrated strong performance, they still suffer from the maliciously crafted adversarial examples. Initially, studies like "Trick Me If You Can" [57] primarily relied on human annotators to construct effective adversarial question-answering examples. This methodology, however, inherently constrained scalability and increased resource demands. As research progressed, automated approaches for attacking textual classifiers in QA models emerged. Gradient-based methods, as employed in RobustQA [63], UAT [56], and HotFlip [10], were developed to identify and modify the most influential words affecting model answers. Building upon a deeper understanding of QA tasks, subsequent studies explored more targeted strategies. For instance, Position Bias [24], TASA [5], and Entropy Maximization [49] investigated the manipulation of sentence locations and the analysis of answer sentences to identify vulnerable parts of the context. These approaches refined the attack methods by applying modifications through paraphrasing or replacing original sentences, thus enhancing the effectiveness of adversarial examples. However, these methods encounter two primary challenges: 1) None of these attack methods is suitable for both "informative queries" and "boolean queries". 2) Constraining the search space for optimal vulnerable words to answer-related sentences compromises attack effectiveness; meanwhile, targeting entire sentences proves inefficient [17].

In addition, Boolean Queries seek a simple binary "Yes" or "No" answer. Models like BERT [7], RoBERTa [70], and GPT variants [22, 3, 1, 51], which excel at sentence-level understanding and token classification, are widely used for Boolean QA tasks. These models leverage their deep contextual understanding of language to accurately determine whether a given statement is true or false, making them state-of-the-art baselines for the task. Researchers have proposed various approaches to target boolean classifiers in the context of Boolean Queries attacks. Attacks like [31, 12, 18, 66, 46], which involve adding, relocating, or replacing words, are based on the influence that

each word has on the prediction. They retrieve word importance by the output confidence to the level or with gradient. However, gradient calculation is computationally intensive and ineffective when dealing with long context input, and knowing victim models' internal information is unrealistic in practice.

We present QA-Attack, an adversarial attack framework tailored for both Informative Queries and Boolean Queries in QA models. QA-Attack uses a Hybrid Ranking Fusion (HRF) algorithm that integrates two methods: Attention-based Ranking (ABR) and Removal-based Ranking (RBR). ABR identifies important words by analyzing the attention weights during question processing, while RBR evaluates word significance by observing changes in the model's output when specific words are removed. The HRF algorithm combines these insights to locate vulnerable tokens, which are replaced with carefully selected synonyms to generate adversarial examples. These examples mislead the QA system while preserving the input's meaning. This unified attack method improves both performance and stealth, ensuring realistic applicability for both types of queries. In summary, our work makes the following key contributions:

- We present QA-Attack with a Hybrid Ranking Fusion (HRF) algorithm designed to target question-answering models. This novel approach integrates attention and removal ranking techniques, accurately locating vulnerable words and fooling the QA model with a high success rate.
- Our QA-Attack can effectively target multiple types of questions. This adaptability allows our method to exploit vulnerabilities across diverse question formats, which significantly broadens the scope of potential attacks in various real-world scenarios.
- QA-Attack generates adversarial examples by implementing subtle word-level changes that preserve both linguistic and semantic integrity while minimizing the extent of alterations, and we conduct extensive experiments on multiple datasets and victim models to thoroughly evaluate our method's effectiveness in attacking QA models.

The rest of this paper is structured as follows. We first review QA system baselines and adversarial attacks for QA models in Section 2. Then we detail our proposed method in Section 3. We evaluate the performance of the proposed method through extensive empirical analysis in Section 4. We conclude the paper with suggestions for future work in Section 5.

2 Related Work

This section provides a comprehensive overview of question-answering models and examines the existing research on adversarial attacks against them.

2.1 Question Answering Models

Question answering represents a complex interplay of NLP, information retrieval, and reasoning capabilities [50, 64]. Basically, these models are designed to process an input question and a context passage, extracting or generating an appropriate answer through elaborate analysis of the semantic relationships between these elements [59].

3

Modern QA systems typically rely on deep learning models with transformer-based architectures like BERT [7] and its variants [48, 70, 26] being particularly prevalent. These models excel at capturing contextual information and understanding nuanced relationships in the text with transformers, allowing them to perform impressively on QA tasks. In addition to these transformer models, encoder-decoder architectures such as T5 [43, 21] and BART [27], GPT [4] and PEGASUS [68] have also become prominent in QA models. These models utilize an encoder to process the input question and context, transforming them into a rich, context-aware representation, and the decoder is then used to generate a coherent and contextually appropriate answer.

2.2 Previous Works on Attacking QA Models

With the development of NLP techniques, recent research has increasingly focused on developing sophisticated textual adversarial examples for QA systems [57]. The inherent differences between "informative queries" and "boolean queries" necessitate distinct attacking diversities due to their unique answer structures [56]. Attacks on boolean QA pairs closely resemble methods used to mislead textual classifiers. These attacks primarily operate at the word level, aiming to manipulate the model's binary (yes/no) output [31, 12]. In contrast, informative queries present a more complex challenge. These attacks frequently target the sentence level, requiring an approach to disrupt the model's comprehensive understanding [28].

2.2.1 Boolean Queries Attacks

Boolean queries are similar to classification tasks in NLP, while the answer is based on two-way input: question and context. They are vulnerable to attacks designed for NLP classifiers when question and context are simply encoded and concatenated. Approaches such as [31], [12], [18], [66], and [46] concentrate on altering individual words based on their influence on model predictions. These methods typically employ carefully selected synonyms for word substitution. The process of word replacement is guided either by the direct use of BERT Masked Language Model (MLM) [7] or by leveraging gradient information to determine optimal substitution candidates. While effectively fool classifiers (boolean queries), these attacks were initially designed for classification tasks and have shown limited efficacy when applied to the question-and-context format of QA systems. To address this limitation, some attack methods for Seq2Seq models have been adapted for QA models. UAT [56], which averages gradients and modifies input data to maximize the model's loss, has been adapted for QA but still struggles with boolean queries due to their simplicity. Similarly, TextBugger [29], which focuses on character-level perturbations, also faces challenges in handling the deeper semantic understanding required in QA, especially for multi-sentence reasoning. Liang's approach [32], relying on confidence-based manipulations, has difficulty reducing the model's certainty in boolean queries where the binary answers leave less room for variation in confidence. Although these approaches offer improved accuracy in attacking informative questions with minor modifications, they struggle with boolean queries. We argue that these methods face challenges in identifying the most vulnerable words when dealing with concatenated question-context input relationships.

The MLQA attack [47] attempts to bridge this gap by utilizing attention weights to identify and alter influential words. However, this method, developed specifically for multi-language BERT models, may not fully address QA-specific vulnerabilities.

2.2.2 Informative Queries Attacks

In contrast to boolean queries, adversarial attacks on informative queries within QA systems share fundamental similarities with attacks on other Seq2Seq models [30, 2, 34], concentrating more on the inter-relationship between question and context. The defense mechanisms like RobustQA [63] have been developed to enhance model resilience through improved training methods, and sophisticated attacks continue to successfully compromise these systems, especially when employing subtle manipulations of key input elements. Character-level attack methods, notably HotFlip [10], have demonstrated significant success by strategically flipping critical characters based on gradient information, leading to misinterpreting informative inputs. In the multilingual domain, MLQA [53] leverages attention weights to identify and target crucial words, though its attention mechanism, primarily designed for multilingual functionality, may not fully exploit the intricate vulnerabilities within the model's attention architecture. Advanced techniques have emerged to target the influence that answers have on QA systems. Position Bias and Entropy Maximization methods exploit model weaknesses by manipulating contextual patterns and answer positioning, particularly effective in scenarios involving complex, lengthy responses. Syntactically Controlled Paraphrase Networks (SCPNs) [16] generate adversarial examples through strategic syntactic alterations while preserving semantic meaning. TASA (Targeted Adversarial Sentence Analysis) [5] primarily relies on manipulating the answer sentences to mislead QA models, making it particularly effective for informative queries where complex responses provide more opportunities for subtle modifications. However, this approach is not suitable for boolean queries, as the simplicity of yes/no answers limits the sentence-level manipulations that TASA depends on.

3 Our Proposed Attack Method

In this section, we introduce the QA-Attack algorithm. It can be summarized into three main steps. First, the method effectively captures important words in context by processing pairs of questions and corresponding context using attention-based and removal-based ranking approaches. Then, attention and removal scores are combined, allowing the identification of the most influential words. At last, a masked language model [7] is utilized to identify potential synonyms that could replace the targeted words. The overall workflow of QA-Attack is shown in Figure 1. In the following sections, we explain our model in detail.

3.1 Problem Setting

Given a pre-trained question-answering model F, which receives an input of context C, question q, and outputs answer a, such that $F(q, C) = a$. The objective is to deceive the performance of F with perturbed context C' such that $F(q, C') \neq a$. To craft C',
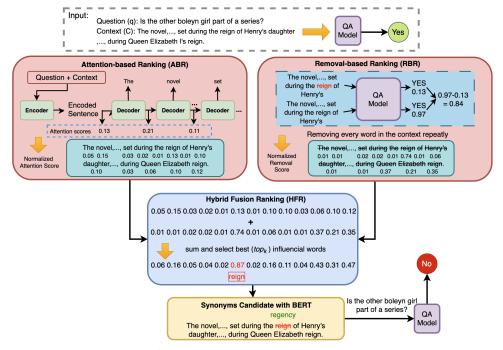
Fig. 1 The workflow of our QA-Attack algorithm for QA models. It processes question-context pairs through two parallel modules: Attention-based Ranking (ABR) and Removal-based Ranking (RBR). These modules generate attention and removal scores respectively for each word using customized attention mechanisms and removal ranking strategies. The scores are then aggregated, and the top_k highest-scoring words are selected as candidates. Finally, these candidates are replaced with BERT-generated synonyms to create adversarial examples that can effectively mislead the QA model.

a certain number of perturbation c_{adv} is added to the context C by replacing some of its original tokens $\{c_1, c_2, ..., c_n\}$.

3.2 Attention-based Ranking (ABR)

Attention mechanisms were first used in image feature extraction in the computer vision field [62, 61, 11]. However, they were later employed by [2] to solve machine translation problems. In translation tasks, attention mechanisms enable models to prioritize and focus on the most relevant parts of the input data [34]. In question-answering tasks, attention scores are imported to examine the relationships between question and context, allowing the model to determine which words or phrases are most relevant to answering the question [60]. Hence, we leverage the attention score to identify target words for our attack. We employ the attention mechanism from T5 [43] that has been specifically optimized for question-answering tasks in UnifiedQA [21]. As shown in Fig. 1, the "Attention-based Ranking" begins by encoding the input context and question through an encoder. During the encoding process, self-attention allows the model to analyze how each word in the input relates to every other word, effectively highlighting the words that carry the most weight in understanding both question and context. In the decoding process, cross-attention further refines this by

---

Algorithm 1: QA-Attack Algorithm

Input : QA victim model $F(\cdot)$, logits L, question q, context C, words in the context c, reference answer a, attention network A, top k words to attack top_k, number of synonyms d, BERT MLM model $BERT$ for generating synonyms.

Output: Optimal adversarial sample C'

---

1 // **Attention-based Ranking**;
2 Compute attention scores: $\alpha \leftarrow [(c, A(q + C))]$;
3 Initialize attention score list: $attention\_scores \leftarrow [\,]$;
4 **for** *each score in* α **do**
5   **if** *score* $\in C$ **then**
6   Append *score* to *attention_scores*;
7   **end**
8 **end**
9 // **Removal-based Ranking**;
10 Initialize importance score list: $importance\_scores \leftarrow [\,]$;
11 **for** *each c in* C **do**
12   Generate modified context: $C^* \leftarrow C$ excluding c;
13   Compute importance score:
      $importance\_scores.append(|F(q, C^*) - F(q, C)|)$;
14 **end**
15 // **Hybrid Ranking Fusion**;
16 Combine attention and importance scores:
   $combined\_scores \leftarrow attention\_scores \cup importance\_scores$;
17 Select top_k words: $top\_k\_list \leftarrow \text{sort}(combined\_scores)[\,: top_k]$;
18 Initialize adversarial examples list: $Adv\_list \leftarrow [\,]$;
19 **for** *each t in* top_k_list **do**
20   Generate adversarial token from d potential synonyms: $c_{adv} \leftarrow BERT(t)$;
21   Create adversarial context: $\Delta C \leftarrow [c_1, \ldots, c_{adv}, \ldots, c_n]$;
22   Append ΔC to Adv_list;
23 **end**
24 Initialize maximum gap: max_gap $\leftarrow -\infty$;
25 Initialize optimal adversarial context: $C' \leftarrow \emptyset$;
26 **for** *each adv in* Adv_list **do**
27   **if** $F(adv) \neq a$ **then**
28   Compute gap: gap $\leftarrow L(F(adv)) - L(F(C))$;
29   **if** $gap > max\_gap$ **then**
30   Update maximum gap: max_gap $\leftarrow$ gap;
31   Update optimal adversarial context: $C' \leftarrow adv$;
32   **end**
33   **end**
34 **end**
35 **return** Optimal adversarial sample C'

---

focusing on the parts of the input most relevant to generating the correct output. By averaging the attention scores of all layers and heads, we match them to each input word.

The implement details are shown in Algorithm 1. The question & context pair is fitted into attention network A, and we filter out the attention scores for context (lines 1 to 8 of Algorithm 1). Then, the attention score of each word corresponding to each layer is summed up. After averaging and normalization, the word-level attention score is obtained.

3.3 Removal-based Ranking (RBR)

Previous studies on adversarial attacks in the text have shown that each word's significance can be quantified using an importance score [18, 30, 5, 31]. This score is largely determined by how directly the word influences the final answer. To enhance the efficacy of ranking progress, we rank each word in the context to obtain the removal importance score (lines 9 to 14 of Algorithm 1). Given the input context C containing n words from c_1 to c_n and question q, the importance score (removal score) of the $i\,th$ $(1 \leq i \leq n)$ word c_i is:

$$I_i = L_F(a \mid q, C) - L_F(a \mid q, C \setminus c_i), \tag{1}$$

where $C \setminus c_i$ represents the context after deleting c_i, and $L_F = \log P(a \mid q, C)$ refers to the probability (logits) of the label, respectively.

3.4 Hybrid Ranking Fusion (HRF)

The attention-based and removal-based word selection techniques offer complementary perspectives on token significance, each highlighting different aspects of word importance. Consequently, we tend to choose words that both methods consider significant. This is achieved by adding the scores from each method for every word to create a fusion score.

When generating a fusion score, we address several key factors. First, we independently normalize the attention and removal scores before adding them together. Then, to balance attack effectiveness and efficiency, we introduce a top_k parameter, a positive integer that controls the number of words targeted. Finally, we select the top_k highest-scoring words for modification (lines 15 to 18 of Algorithm 1).

3.5 Synonym Selection

Various synonym generation methods exist, including Word2Vec [36], Hownet [9], and WordNet [9]. We adopt BERT [7] for synonym selection due to its textual capabilities, which enable it to generate synonyms based on the complete sentence structure. Unlike Word2Vec's static embeddings or WordNet's fixed synonym lists, BERT's context-sensitive approach allows for dynamic synonym selection that preserves both semantic meaning and grammatical correctness. This contextual awareness makes BERT particularly effective for crafting natural and semantically coherent adversarial examples.

We process each selected word in the context by replacing it with the "[MASK]" token. This modified context is then input into the BERT Masked Language Model (MLM) to predict the most likely substitutions for the masked word. To expand the range of potential samples, we introduce a parameter d that controls the number of synonym substitutions considered (lines 19 to 23 of Algorithm 1). This approach allows us to generate a diverse set of imperceptible replacements while maintaining contextual relevance.

3.6 Candidate Selection

We define an optimal adversary as one that maximizes the difference between the predicted answer and the attacked answer. For boolean queries, following textual classifier approaches that utilize logits to decide output label (yes/no), we compare the logits of output answers. For informative queries, we sum the logits of individual words. Using L to denote the logits derivation function, we identify the optimal adversary from the "Adv_list" as shown in lines 24 to 35 of Algorithm 1.

4 Experiment and Analysis

In this section, we present a comprehensive evaluation of QA-Attack's performance compared to current state-of-the-art baselines. Our analysis covers several key aspects with various metrics, providing a thorough understanding of our method's capabilities, limitations, and performance across diverse scenarios. We provide a detailed analysis of attack performance and imperceptibility (Sec. 4.4). Besides, to gain deeper insights, we conduct ablation studies (Sec. 4.5) and assess attacking efficiency (Sec. 4.6). In addition, we examine QA-Attack's response to defense strategies (Sec. 4.8), exploring the effects of adversarial retraining (Sec. 4.7) and investigating the transferability of attacks (Sec. 4.9). Additionally, we report the preference of our attack by investigating parts of speech preference (Sec. 4.10) and analyzing its robustness versus the scale of pre-trained models (Sec. 4.11).

4.1 Experiment Settings and Evaluation Metrics

The base setting of our experiments is let $top_k = 5$, $d = 2$, and use a BERT-base-uncased[1] with 12 Transformer encoder layer (L) and 768 hidden layers (H) as the synonym generation model. Tables 3, 4, and 5 summarize the experimental results on informative queries datasets, offering a comparative analysis of our QA-Attack method against five state-of-the-art QA baselines. For boolean queries, we present the attacking results on the BoolQ dataset in Table 6. Some visualised examples are shown in Table 2. Besides, we provide code for the reproductivity of our experiments[2]. The metrics used in our experiment are:

- **F1**: The F1 score balances precision and recall, providing a nuanced view of how much the attacked answers match reference answers.

---

[1] https://github.com/google-research/bert/?tab=readme-ov-file.
[2] Our code is available at: https://github.com/UTSJiyaoLi/QA-Attack.

Table 1 Dataset distribution and corresponding baseline performance (F1).

Dataset	Data Distribution				Model Performance (F1)		
	Total	Train	Validation	Test	T5	LongT5	BERT$_{base}$
SQuAD 1.1	100,000	87,600	10,570	N/A	88.9	89.5	88.5
SQuAD V2.0	150,000	130,319	11,873	N/A	81.3	83.2	74.8
NewsQA	119,000	92,549	5,165	5,126	66.8	67.2	60.1
BoolQ	16,000	9,427	3,270	3,245	85.2	86.1	80.4
NarrativeQA	45,000	32,747	3,461	10,557	67.5	68.9	62.1

- **ROUGE and BLEU**: A higher BLEU [41] or ROUGE [33] score in context indicates that the adversarial context retains more of the exact phrasing, contributing to better linguistic fluency and coherence.
- **Exact Match (EM)** Measures the percentage of model predictions that exactly match the correct answers in both content and format.
- **Similarity (SIM)**: Evaluates the semantic similarity between original and adversarial context using BERT [7] embeddings. (Note: In our following experiments, EM and SIM are not only measured answers but also reflect the quality of the generated context in Sec. 4.5.3).
- **Modification Rate (Mod)**: Mod measures the proportion of altered tokens in the text. This metric considers each instance of replacement, insertion, or deletion as a single token modification.
- **Grammar Error (GErr)**: GErr measures the increase in grammatical inaccuracies within successful adversarial examples relative to the original text. This measurement employs LanguageTool [39] to enumerate grammatical errors.
- **Perplexity (PPL)**: PPL serves as an indicator of linguistic fluency in adversarial examples [20, 66]. The perplexity calculation utilizes a GPT-2 model with a restricted vocabulary [42].

4.2 Datasets and Victim Models

We assess QA-Attack using four informative queries datasets: SQuAD 1.1 [45], SQuAD V2.0 [44], NarrativeQA [25], and NewsQA [55], along with the boolean queries dataset BoolQ [6].

- SQuAD 1.1: Questions formulated by crowd workers based on Wikipedia articles. Answers are extracted as continuous text spans from the corresponding passages.
- SQuAD 2.0: Extension of SQuAD 1.1 incorporating unanswerable questions. These questions are designed such that no valid answer can be located within the provided passage, adding complexity to the task.
- NarrativeQA: Questions based on entire books or movie scripts. Answers are typically short and abstractive, demanding deeper comprehension and synthesis of narrative elements.
- NewsQA: Questions based on CNN news articles designed to test reading comprehension in the context of current events and journalistic writing.

- BoolQ: Dataset of boolean (yes/no) questions derived from anonymized, aggregated queries submitted to the Google search engine, reflecting real-world information-seeking behaviour.

Our experiment includes three question-answering models for comparison. They are T5[21], LongT5 [14], and BERT$_{base}$ [7]. The LongT5 is an extension of T5 with an encoder-decoder specifically for long contextual inputs. The BERT-based models are structured with bidirectional attention, meaning each word in the input sequence contributes to and receives context from both its left and right sides. Table 1 presents the distribution of dataset splits and F1 scores reported on each QA baseline.

4.3 Baseline Attacks

For our experimental baselines, we employ five leading attack methods: TASA [5], RobustQA [63], Tick Me If You Can (TMYC) [57], T3 [58], and TextFooler [18]. We utilize the official implementation of T3 in its black-box setting, while TASA, TMYC, and RobustQA are employed with their standard configurations. TextFooler, originally not designed for question-answering tasks, was adapted for our experiments. We modified it to process the context only (questions are removed).

4.4 Experiment Analysis

Our experimental results in Table 3, 4, 5 demonstrate that QA-Attack consistently outperforms baseline methods across all informative datasets. As shown in Table 6, our method achieves superior performance on the boolean dataset, surpassing all baseline approaches in degrading victim models' accuracy (note that TASA is designed only for informative queries; it is incompatible with boolean query attacks). For informative queries, comparing performance on attacking LongT5 with SQuAD 1.1 and NarrativeQA datasets (representing shortest and longest contexts) in Table 5, we observe that while F1 and EM scores decrease for longer contexts, QA-Attack maintains superiority over baselines. This indicates our approach's robustness and adaptability to varying context lengths, particularly in long text. The improved performance in longer contexts suggests our HRF approach effectively identifies and targets vulnerable tokens. Regarding semantic consistency, QA-Attack achieves lower similarity scores compared to baseline methods, indicating that the answers generated after the attack deviate more in meaning from the ground truth responses.

Additionally, the quality of the generated adversarial samples is evident from the ROUGE and BLEU scores. Our method consistently achieves higher ROUGE and BLEU scores compared to the baselines, which suggests that the adversarial examples generated by QA-Attack are not only effective in terms of altering the model's output but also maintain a high degree of contextual and linguistic coherence. This is largely due to our synonym selection method, which ensures the replacements are contextually appropriate and semantically relevant. Moreover, the token-level replacement strategy, which only mods fewer words (typically five in the base setting), further ensures that the adversarial examples remain similar to the original context while fooling the model.

11

Table 2 Comparison of original and adversarial contexts for two types of queries. The table highlights the differences between the original and adversarial contexts, as well as the corresponding answers provided by the model before and after the attack.

Question	Was the movie "The Strangers" based on a true story?
Context	The Strangers is a 2008 American slasher film written and directed by Bryan Bertino. Kristen (Liv Tyler) and James (Scott Speedman) are expecting a relaxing weekend at a family vacation home, but their stay turns out to be anything but peaceful as three masked torturers leave Kristen and James struggling for survival. Writer-director Bertino was inspired by real-life events: the Manson family Tate murders, a multiple homicide; the Keddie Cabin Murders, that occurred in California in 1981; and a series of break-ins that occurred in his own neighborhood as a child.
Adversary	The Strangers is a 2008 American slasher thriller written and directed by Bryan Bertino. Kristen (Liv Tyler) and James (Scott Speedman) are spending a relaxing weekend at a family vacation home, but their stay turns out to be anything but peaceful as three masked torturers leave Kristen and James struggling for survival. Writer-director Bertino was influenced by real-life incidents: the Manson family Tate murders, a multiple homicide; the Keddie Cabin Murders, that occurred in California in 1981; and a series of break-ins that occurred in his own home as a child.
Original Answer	Yes
Attacked Answer	No
Question	Who ruled the Duchy of Normandy?
Context	The Normans were famed for their martial spirit ... The Duchy of Normandy, which they formed by treaty with the French crown, was a great fief of medieval France, and under Richard I of Normandy was forged into a cohesive and formidable principality in feudal tenure ... Norman adventurers founded the Kingdom of Sicily ... an expedition on behalf of their duke, William the Conqueror, led to the Norman conquest of England at the Battle of Hastings in 1066.
Adversary	The Normans were famed for their warrior spirit ... The Duchy of Normandy, which they formed by treaty with the French crown, was a great fief of medieval France, and under William I of Normandy was forged into a cohesive and formidable principality in feudal tenure ... Norman adventurers invaded the Kingdom of Sicily ... an invasion on behalf of their duke, William the Conqueror, led to the Norman conquest of England at the siege of Hastings in 1066.
Original Answer	The French crown
Predicted Answer	William I of Normandy

4.5 Ablation and Hyperparameters Studies

To comprehensively validate the efficacy of the proposed QA-Attack method, this section conducts a detailed ablation study, dissecting each component to assess its individual impact and overall contribution to the method's performance.

4.5.1 Effectiveness of Hybrid Fusion Ranking on Multiple Question Types

We test how HRF, ABR, and RBR methods perform across different top_k values on the SQuAD and BoolQ datasets, with d remaining, shown in Fig. 2. HRF consistently

Table 3 Comparative analysis of QA-Attack and baseline models on T5. Drops of BLEU and ROUGE scores (uni-gram) on contexts are reported in the table, with higher values indicating better performance. For F1, EM, and SIM (i.e., similarity) metrics on answers, lower values indicate better performance.

Datasets	Methods	F1↓	EM↓	ROUGE↑	BLEU↑	SIM↓
SQuAD 1.1	TASA [5]	9.21	7.49	89.12	82.88	6.38
	TMYC (Tick Me If You Can) [57]	7.28	8.21	81.91	78.72	8.22
	RobustQA [63]	5.89	7.52	84.23	77.41	6.03
	TextFooler [18]	10.6	10.49	83.11	76.05	6.29
	T3 [58]	5.41	6.29	86.83	73.82	7.23
	QA-Attack (ours)	**4.67**	**5.68**	**90.51**	**84.11**	**5.91**
SQuAD V2.0	TASA [5]	20.09	19.31	70.21	76.06	7.29
	TMYC (Tick Me If You Can) [57]	17.23	20.68	65.19	69.82	9.05
	RobustQA [63]	16.37	18.73	67.71	63.19	8.14
	TextFooler [18]	21.69	24.5	65.33	65.01	9.32
	T3 [58]	11.19	19.68	69.71	73.53	8.82
	QA-Attack (ours)	**9.13**	**15.41**	**72.76**	**77.28**	**6.33**
Narrative QA	TASA [5]	11.79	15.25	68.11	70.36	6.11
	TMYC (Tick Me If You Can) [57]	12.73	9.32	65.91	67.22	7.61
	RobustQA [63]	10.01	13.91	67.19	64.11	6.81
	TextFooler [18]	14.72	18.61	63.85	62.82	11.74
	T3 [58]	11.74	11.37	62.34	60.17	6.28
	QA-Attack (ours)	**5.61**	**7.23**	**69.18**	**75.73**	**5.23**
NewsQA	TASA [5]	8.56	29.44	77.28	69.44	7.11
	TMYC (Tick Me If You Can) [57]	6.12	31.23	77.96	72.49	9.22
	RobustQA [63]	5.12	29.48	83.81	79.82	10.84
	TextFooler [18]	9.01	30.86	74.21	57.44	27.91
	T3 [58]	6.21	28.52	75.22	72.56	14.27
	QA-Attack (ours)	**3.61**	**24.42**	**78.85**	**82.83**	**8.92**

outperforms ABR and RBR for all top_k values on both datasets. This suggests combining attention-based and removal-based ranking in HRF is more effective at generating robust adversarial examples than using either method alone. The graph also shows that as top_k increases, all methods improve, indicating that higher top_k values help identify vulnerable tokens better and lead to more effective attacks.

Despite the better performance at higher top_k values, the study uses $top_k = 5$ as a base setting. This choice balances effectiveness with minimal text modification, ensuring that adversarial examples remain close to the original context while still being effective. The consistent trend across both SQuAD and BoolQ datasets demonstrates that HRF's superior performance holds true for different question types, showing its versatility in attacking various question-answering models. This analysis highlights the practical effectiveness of the HRF method and its ability to generate impactful adversarial examples across different QA tasks.

4.5.2 Effectiveness of Synonyms Selection

To evaluate our Synonyms Selection approach, we conduct comparisons in two aspects. We first compare our BERT-based synonym generation against two alternative methods: WordNet [37], an online database that contains sets of synonyms, and HowNet [9],

Table 4 Comparative analysis of QA-Attack and baseline models on Bert$_{base}$. Drops of BLEU and ROUGE scores (uni-gram) on contexts are reported in the table, with higher values indicating better performance. For F1, EM, and SIM (i.e., similarity) metrics on answers, lower values indicate better performance.

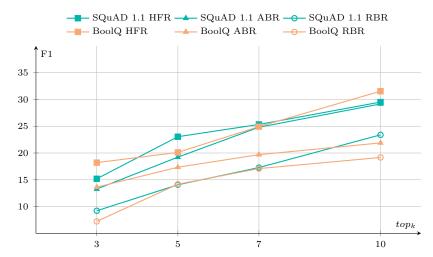| Datasets	Methods	F1↓	EM↓	ROUGE↑	BLEU↑	SIM↓
SQuAD 1.1	TASA [5]	15.27	34.33	82.87	67.22	8.19
	TMYC (Tick Me If You Can) [57]	12.89	28.63	81.51	76.39	10.24
	RobustQA [63]	15.72	25.38	79.28	73.27	15.81
	TextFooler [18]	23.04	37.28	67.28	49.49	14.11
	T3 [58]	8.79	16.11	57.19	63.81	16.92
	QA-Attack (ours)	**6.42**	**13.31**	**91.22**	**77.16**	**7.43**
SQuAD V2.0	TASA [5]	31.22	28.9	77.06	69.05	8.22
	TMYC (Tick Me If You Can) [57]	29.38	27.77	73.81	67.23	10.34
	RobustQA [63]	27.64	31.82	75.67	71.42	11.23
	TextFooler [18]	36.8	29.49	67.14	62.67	13.28
	T3 [58]	26.16	27.47	74.94	70.14	7.24
	QA-Attack (ours)	**22.18**	**21.5**	**80.12**	**75.23**	**4.11**
Narrative QA	TASA [5]	12.11	14.51	61.15	63.04	7.32
	TMYC (Tick Me If You Can) [57]	8.41	10.23	52.89	69.82	10.72
	RobustQA [63]	7.24	9.43	63.81	67.43	9.53
	TextFooler [18]	13.74	18.79	56.11	56.82	14.21
	T3 [58]	8.49	15.35	65.48	67.09	7.83
	QA-Attack (ours)	**3.86**	**9.34**	**69.44**	**71.15**	**5.61**
NewsQA	TASA [5]	16.85	20.95	68.74	69.12	15.22
	TMYC (Tick Me If You Can) [57]	15.86	31.23	77.96	72.49	9.22
	RobustQA [63]	17.72	29.48	83.81	79.82	10.84
	TextFooler [18]	24.13	22.63	59.17	61.22	31.07
	T3 [58]	21.22	22.57	65.14	67.11	18.27
	QA-Attack (ours)	**14.91**	**20.20**	**70.04**	**74.87**	**9.22**

Fig. 2 F1 score analysis for HFR, ABR, and RBR variants of QA-Attack using different top_k values, tested on datasets SQuAD 1.1 and BoolQ.

Table 5 Comparative analysis of QA-Attack and baseline models on LongT5. Drops of BLEU and ROUGE scores (uni-gram) on contexts are reported in the table, with higher values indicating better performance. For F1, EM, and SIM (i.e., similarity) metrics on answers, lower values indicate better performance.

Datasets	Methods	F1↓	EM↓	ROUGE↑	BLEU↑	SIM↓
SQuAD 1.1	TASA [5]	10.61	22.45	80.67	70.41	11.88
	TMYC (Tick Me If You Can) [57]	12.43	29.81	75.37	63.83	13.22
	RobustQA [63]	17.22	31.11	73.11	68.29	17.64
	TextFooler [18]	35.31	44.09	57.77	49.49	25.33
	T3 [58]	9.33	24.52	49.23	60.33	20.87
	QA-Attack (ours)	**7.38**	**18.78**	**84.22**	**72.67**	**9.67**
SQuAD V2.0	TASA [5]	30.71	30.11	64.71	67.28	9.32
	TMYC (Tick Me If You Can) [57]	34.11	33.88	64.21	65.11	14.82
	RobustQA [63]	29.01	39.59	62.91	68.22	13.09
	TextFooler [18]	38.25	34.67	60.47	64.16	15.44
	T3 [58]	30.44	30.13	65.81	63.72	8.29
	QA-Attack (ours)	**27.11**	**24.73**	**77.37**	**70.32**	**5.29**
Narrative QA	TASA [5]	8.22	10.67	69.83	65.77	9.53
	TMYC (Tick Me If You Can) [57]	9.36	11.33	63.15	64.27	14.72
	RobustQA [63]	15.83	12.03	64.28	63.12	12.77
	TextFooler [18]	12.77	14.82	62.99	54.21	17.33
	T3 [58]	8.38	8.26	63.92	66.32	8.92
	QA-Attack (ours)	**4.62**	**5.33**	**70.33**	**68.32**	**7.44**
NewsQA	TASA [5]	16.85	24.54	64.83	66.81	14.82
	TMYC (Tick Me If You Can) [57]	19.28	29.01	62.88	68.67	11.43
	RobustQA [63]	17.23	27.42	58.32	57.22	13.37
	TextFooler [18]	27.22	26.39	53.33	53.01	25.82
	T3 [58]	17.83	25.87	63.25	65.43	19.27
	QA-Attack (ours)	**15.32**	**24.12**	**68.23**	**70.55**	**10.48**

which produces semantically similar words using its network structure. Using the base configuration, we evaluate the EM scores when attacking T5 and BERT$_{base}$ models across three datasets: SQuAD 1.1, NarrativeQA, and BoolQ. The results demonstrate that our QA-Attack with BERT$_{base}$ consistently achieved superior performance compared to other methods across all datasets and victim models.

On the other hand, we also examine the impact of parameter d in Synonym Selection, which determines the number of synonyms obtained from the Masked Language Model (MLM). Table 8 illustrates that as d increases from 1 to 3, F1 scores consistently decrease across all datasets, indicating improved attack performance. This trend suggests that a more aggressive setting (higher d) is more effective in compromising model accuracy across various datasets.

4.5.3 Texual Quality of Word Candidates

In our ablation study, detailed in Table 9, we investigate the quality of adversarial examples generated by various attack methods on the T5 model using the SQuAD 1.1 dataset. We evaluate our word replacement technique with encoder-decoder candidate generation (T3), as well as sentence-level modification methods (TASA, TMYC). The results indicate that our word-level synonym selection approach outperformed all other

Table 6 Attack performance comparison on baseline models using the BoolQ dataset, with top results highlighted in bold. Note that TASA [5] is not applicable to boolean questions.

Victim Models	Methods	F1↓	EM↓	ROUGE↑	BLEU↑	SIM↓
UnifiedQA	TASA [5]	—	—	—	—	—
	TMYC (Tick Me If You Can) [57]	17.43	19.36	82.09	77.23	21.83
	RobustQA [63]	14.33	18.92	79.15	80.33	13.22
	TextFooler [18]	20.11	19.07	80.91	83.25	33.82
	T3 [58]	15.16	14.74	71.32	68.79	15.82
	QA-Attack (ours)	**8.64**	**13.9**	**87.31**	**86.57**	**11.42**
Bert$_{base}$	TASA [5]	—	—	—	—	—
	TMYC (Tick Me If You Can) [57]	21.35	13.28	63.21	70.57	7.34
	RobustQA [63]	24.81	9.21	69.22	76.01	6.67
	TextFooler [18]	33.02	11.57	65.11	67.81	8.17
	T3 [58]	22.06	11.02	76.17	74.62	6.23
	QA-Attack (ours)	**18.39**	**6.51**	**77.21**	**78.11**	**4.66**
LongT5	TASA [5]	—	—	—	—	—
	TMYC (Tick Me If You Can) [57]	29.77	9.82	67.04	73.22	7.43
	RobustQA [63]	24.56	8.21	70.49	71.83	9.33
	TextFooler [18]	33.02	11.57	65.11	67.81	8.17
	T3 [58]	22.06	11.02	76.17	74.62	6.23
	QA-Attack (ours)	**18.39**	**6.51**	**77.21**	**78.11**	**4.66**

Table 7 EM scores for attacks on T5 and BERT$_{base}$ models using three distinct synonym generation methods. Lower scores indicate more effective attacks.

Methods	Victim Models	Datasets		
		SQuAD 1.1	NarrativeQA	BoolQ
HowNet	T5	14.22	7.25	29.08
	BERT$_{base}$	7.66	4.52	26.91
WordNet	T5	5.31	3.99	21.63
	BERT$_{base}$	7.23	5.67	19.35
BERT$_{base}$ (ours)	T5	**4.67**	**5.61**	**8.64**
	BERT$_{base}$	**6.42**	**3.86**	**18.39**

Table 8 F1 scores demonstrating QA-Attack's performance across five datasets under different d values (i.e., number of synonym candidates for substitutions).

	SQuAD 1.1	SQuAD V2.0	BoolQ	NarrativeQA	NewQA
$d = 1$	8.52	14.72	19.22	7.63	10.66
$d = 2$	4.67	9.13	15.16	5.61	3.61
$d = 3$	2.17	7.26	11.43	3.71	3.27

baselines. Notably, our word-level attack maintains a lower grammar error rate and higher linguistic fluency than alternative methods. Although RobustQA employs the same synonym selection strategy, it requires more word modifications to successfully attack the model and tends to produce more adventurous alterations.

Table 9 Performance metrics for different word candidate selection strategies against T5 model on SQuAD 1.1 dataset.

Methods	EM↓	SIM↑	Mod↓	PPL↓	GErr↓
TASA [5]	9.21	6.38	8.15	143	0.13
TMYC (Tick Me If You Can) [57]	7.28	8.22	9.21	151	0.14
RobustQA [63]	5.89	6.03	8.35	147	0.15
T3 [58]	5.41	7.23	7.93	133	0.13
TextFooler [18]	10.60	6.29	8.17	136	0.14
QA-Attack (ours)	**5.68**	**5.91**	**7.24**	**125**	**0.12**

Table 10 Time consumption (seconds per sample) for various methods and datasets. A lower value indicates better performance.

	NarrativeQA	SQuAD 1.1	SQuAD V2.0	NewsQA	BoolQ
TASA [5]	28.77	15.82	18.25	10.72	–
TMYC (Tick Me If You Can) [57]	25.61	12.75	16.33	9.21	7.42
RobustQA [63]	25.82	24.46	22.15	12.81	15.82
T3 [58]	26.52	21.37	28.38	14.74	7.93
QA-Attack (ours)	**23.51**	**10.61**	**12.38**	**8.32**	**7.22**

4.6 Platform and Efficiency Analysis

In this section, we evaluate QA-Attack's computational efficiency under base settings. We measure efficiency using time consumption per sample, expressed in seconds, where a lower value indicates superior performance. As shown in Table 10, the outcomes reveal that QA-Attack exhibits remarkable time efficiency, consistently outperforming baseline methods across both long-text (NarrativeQA) and short-text (SQuAD 1.1) datasets. This superior performance can be attributed to QA-Attack's innovative Hybrid Ranking Fusion (HRF) strategy, which effectively identifies vulnerable words within the text, significantly enhancing the speed of the attack process.

4.7 Adversarial Retraining

In this section, we investigate QA-Attack's potential for enhancing downstream models' accuracy. We employ QA-Attack to generate adversarial examples from SQuAD 1.1 training sets and incorporate them as supplementary training data. We reconstruct the training set with varying proportions of adversarial examples added to the raw training set. The retraining process with this augmented data aims to examine how test accuracy changes in response to the inclusion of adversarial examples. As illustrated in Fig. 3, re-training with adversarial examples slightly improves model performance when less than 30% of the training data consists of adversaries. However, performance decreases when the proportion of adversaries exceeds 30%. This finding indicates that the optimal ratio of adversarial examples in training data needs to be determined empirically, which aligns with conclusions from previous attacking methods. To evaluate how re-training helps defend against adversarial attacks, we analyze the robustness of T5 models trained with varying proportions of adversarial examples (0%, 10%, 20%, 30%, 40%) from different attack methods, as shown in
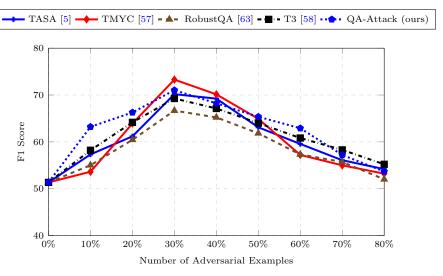
Fig. 3 The performance of the T5 model re-trained on the SQuAD 1.1 dataset with mixed adversarial examples generated by TASA [5], TMYC [57], RobustQA [63], T3 [58], and our QA-Attack.

Fig. 3. A lower F1 score indicates higher model susceptibility to adversarial attacks. The attack performance of the re-trained model is shown in Fig. 4. It demonstrates that incorporating adversarial examples during training consistently improves model robustness, as evidenced by increasing F1 scores across all attack methods. Notably, QA-Attack emerges as the most effective approach, consistently outperforming other methods, with its advantage becoming particularly pronounced at higher percentages of adversarial training data.

4.8 Attacking Models with Defense Mechanism

Defending NLP models against adversarial attacks is crucial for maintaining the reliability of language processing systems in real-world applications [13]. To further analyse how attacks are performed under defense systems, we deploy two distinct defense mechanisms to investigate our attack performance under defense systems. The first is Frequency-Guided Word Substitutions (FGWS) approach [38], which excels at detecting adversarial examples. The second is Random Masking Training (RanMASK) [67], a technique that enhances model robustness through specialized training procedures. We perform the adversarial attack on T5 on datasets SQuAD 1.1, NarrativeQA and BoolQ, and the results are presented in Table 11. The results show that QA-Attack demonstrates superior adversarial robustness across multiple benchmark datasets, consistently outperforming existing methods against state-of-the-art defenses.

4.9 Transferability of Attacks

To evaluate our model's transferability, we test the adversarial samples generated for T5 on three distinct question-answering models: RoBERTa [70], DistilBERT [48], and MultiQA [53]. We also compare the transferability of three baseline methods: TASA,
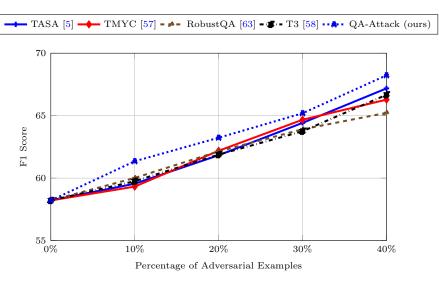
18

Fig. 4 F1 scores of attacking T5 models retrained with increasing proportions of adversarial examples generated by baseline methods (TASA [5], TMYC [57], RobustQA [63], T3 [58]) and our QA-Attack.

Table 11 Effectiveness of defense mechanisms (FGWS [38] and RanMASK [67]) against QA-Attack: EM scores of T5 model output answers across SQuAD 1.1, NarrativeQA, and BoolQ datasets. Lower scores indicate higher attack success against defenses.

Datasets	Defense	TASA	RobustQA	TMYC	T3	QA-Attack
SQuAD 1.1	FGWS [38]	34.71	39.42	28.51	24.11	**21.03**
	RanMASK [67]	32.17	39.78	44.81	41.09	**30.26**
NarrativeQA	FGWS [38]	49.28	44.62	**37.21**	45.17	38.33
	RanMASK [67]	38.41	37.14	41.62	43.81	**34.47**
BoolQ	FGWS [38]	45.71	47.37	38.97	45.33	**38.34**
	RanMASK [67]	41.63	42.88	47.25	42.17	**40.51**

TextFooler, and T3, under identical experimental conditions. As shown in Fig. 5, QA-Attack effectively degrades other QA models' performance on both the NarrativeQA and BoolQ datasets. This suggests that the transferring attack performance of our QA-Attack consistently outperforms the baselines.

4.10 Parts of Speech Preference

To further understand the candidate words' distribution of our word-level attack, we examine its attacking preference in terms of Parts of Speech (POS), highlighting vulnerable areas within the input context. We use the Stanford POS tagger [54] to label each attacked word, categorizing them as *noun, verb, adjective (Adj.), adverb (Adv.)*, and *others (e.g., pronoun, preposition, conjunction)*. Table 12 illustrates the POS preference of our QA-Attack compared to baseline methods in the base setting. For "informative queries" on SQuAD dataset, most attacking methods predominantly target *nouns*, while TASA shows a slight preference for *adverbs*. In the case of "boolean
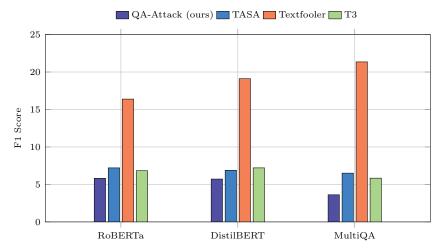
Fig. 5 F1 scores for transfer attacks on three other QA models using adversarial samples generated for UnifiedQA. A lower value indicates better performance.

Table 12 POS preference with respect to choices of victim words among attacking methods.

Datasets	Methods	Noun (%)	Verb (%)	Adj. (%)	Adv. (%)	Others (%)
SQuAD 1.1	TASA	*35*	12	13	**36**	4
	TMYC	**47**	*21*	11	5	17
	RobustQA	**34**	13	*22*	16	15
	Textfooler	**44**	13	*23*	8	12
	T3	**60**	*17*	6	7	10
	QA-Attack	*34*	9	18	3	**36**
BoolQ	TASA	–	–	-	–	-
	TMYC	14	*19*	12	**35**	20
	RobustQA	*19*	14	**27**	23	17
	Textfooler	**41**	15	*27*	7	10
	T3	**42**	*13*	20	16	9
	QA-Attack	10	19	*25*	18	**28**

queries" on BoolQ dataset, all methods frequently focus on *adjectives* and *adverbs*. Notably, our QA-Attack demonstrates a higher preference for the "others" category. Given that these parts of speech (pronouns, prepositions, and conjunctions) carry limited semantic content, we suggest that altering them may not significantly affect the linguistic or semantic aspects of prediction. However, such modifications could disrupt sequential dependencies, potentially compromising the contextual understanding of QA models and misleading their answers.

4.11 Robustness versus the Scale of Pre-trained Models

From the attacking results in Table 4 discussed in Sec 4.4, we recognize the limitation of our QA-Attack on $BERT_{base}$, with $L = 12$ and $H = 768$, which does not sufficiently support robust experimental outcomes. To address this issue and gain more

Table 13 A comparative analysis to attacking various sizes of BERT model on SQuAD 1.1 dataset. A lower value indicates better attack performance.

Versions	BERT tiny	BERT mini	BERT medium	BERT large
Size	L = 2, H = 128	L = 4, H = 256	L = 8, H = 512	L = 24, H = 1024
EM $\downarrow$	11.82	13.26	13.31	14.25
F1 $\downarrow$	5.67	6.35	6.42	7.24
SIM $\downarrow$	6.23	7.12	7.43	8.38

comprehensive insights, we conducted experiments with four different sizes of BERT [7] models[3]: $BERT_{tiny}$, $BERT_{mini}$, $BERT_{medium}$, and $BERT_{large}$. Our findings, detailed in Table 13, demonstrate a positive correlation between model size and experimental robustness. The effectiveness of adversarial attacks decreases as the complexity and capacity of the BERT model increase, suggesting that deeper architectures provide better protection against adversarial perturbations.

5 Conclusion and Future Work

The robustness of QA models has been increasingly challenged by adversarial attacks. These attacks expose the vulnerabilities of models used in various tasks, including information retrieval, conversational agents, and machine comprehension. To address this, we introduced QA-Attack, which leverages Hybrid Ranking Fusion (HRF) to conduct effective attacks by identifying and modifying the most critical tokens in the input text. Through a combination of attention-based and removal-based ranking strategies, QA-Attack successfully disrupts model predictions while maintaining high levels of semantic and linguistic coherence. Extensive experiments have demonstrated that our method outperforms existing attack techniques regarding attack success, fluency, and consumption across various datasets, confirming its efficacy in undermining the robustness of state-of-the-art QA models.

While adversarial attacks such as QA-Attack highlight the weaknesses in QA systems, they also provide an opportunity to test and improve model robustness. In future work, we plan to focus on developing defence strategies that mitigate these vulnerabilities. Furthermore, we intend to extend QA-Attack to handle more complex and diverse QA scenarios, including multiple-choice questions and multi-hop reasoning [65], to ensure that our method remains a powerful tool for evaluating and improving the robustness of QA systems in an evolving landscape of adversarial threats.

References

[1] Antaki, F., Milad, D., Chia, M.A., Giguère, C.É., Touma, S., El-Khoury, J., Keane, P.A., Duval, R.: Capabilities of gpt-4 in ophthalmology: an analysis of model entropy and progress towards human-level medical question answering. British Journal of Ophthalmology **108**(10), 1371–1378 (2024)

---

[3]Different sizes of BERT models can be obtained from https://github.com/google-research/bert/

[2] Bahdanau, D., Cho, K., Bengio, Y.: Neural machine translation by jointly learning to align and translate. CoRR **abs/1409.0473** (2014)

[3] Bongini, P., Becattini, F., Del Bimbo, A.: Is gpt-3 all you need for visual question answering in cultural heritage? In: European Conference on Computer Vision. pp. 268–281. Springer (2022)

[4] Brown, T., Mann, B., Ryder, N., Subbiah, M., Kaplan, J.D., Dhariwal, P., Neelakantan, A., Shyam, P., Sastry, G., Askell, A., Agarwal, S., Herbert-Voss, A., Krueger, G., Henighan, T., Child, R., Ramesh, A., Ziegler, D., Wu, J., Winter, C., Hesse, C., Chen, M., Sigler, E., Litwin, M., Gray, S., Chess, B., Clark, J., Berner, C., McCandlish, S., Radford, A., Sutskever, I., Amodei, D.: Language models are few-shot learners. In: Larochelle, H., Ranzato, M., Hadsell, R., Balcan, M., Lin, H. (eds.) Advances in Neural Information Processing Systems. vol. 33, pp. 1877–1901. Curran Associates, Inc. (2020)

[5] Cao, Y., Li, D., Fang, M., Zhou, T., Gao, J., Zhan, Y., Tao, D.: TASA: Deceiving question answering models by twin answer sentences attack. In: Goldberg, Y., Kozareva, Z., Zhang, Y. (eds.) Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing. pp. 11975–11992. Association for Computational Linguistics, Abu Dhabi, United Arab Emirates (Dec 2022)

[6] Clark, C., Lee, K., Chang, M.W., Kwiatkowski, T., Collins, M., Toutanova, K.: BoolQ: Exploring the surprising difficulty of natural yes/no questions. In: Burstein, J., Doran, C., Solorio, T. (eds.) Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers). pp. 2924–2936. Association for Computational Linguistics, Minneapolis, Minnesota (Jun 2019)

[7] Devlin, J., Chang, M.W., Lee, K., Toutanova, K.: BERT: Pre-training of deep bidirectional transformers for language understanding. In: Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies. pp. 4171–4186. Association for Computational Linguistics, Minneapolis, Minnesota (June 2019)

[8] Dong, H., Dong, J., Yuan, S., Guan, Z.: Adversarial attack and defense on natural language processing in deep learning: A survey and perspective. In: International conference on machine learning for cyber security. pp. 409–424. Springer (2022)

[9] Dong, Z., Dong, Q.: Hownet-a hybrid language and knowledge resource. In: International conference on natural language processing and knowledge engineering, 2003. Proceedings. 2003. pp. 820–824. IEEE (2003)

[10] Ebrahimi, J., Rao, A., Lowd, D., Dou, D.: HotFlip: White-box adversarial examples for text classification. In: Gurevych, I., Miyao, Y. (eds.) Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers). pp. 31–36. Association for Computational Linguistics, Melbourne, Australia (Jul 2018)

[11] Galassi, A., Lippi, M., Torroni, P.: Attention in natural language processing. IEEE Transactions on Neural Networks and Learning Systems **32**(10), 4291–4308 (Oct 2021)

[12] Garg, S., Ramakrishnan, G.: Bae: Bert-based adversarial examples for text classification. In: Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP). Association for Computational Linguistics (2020)

[13] Goyal, S., Doddapaneni, S., Khapra, M.M., Ravindran, B.: A survey of adversarial defenses and robustness in nlp. ACM Computing Surveys **55**(14s), 1–39 (2023)

[14] Guo, M., Ainslie, J., Uthus, D., Ontanon, S., Ni, J., Sung, Y.H., Yang, Y.: LongT5: Efficient text-to-text transformer for long sequences. In: Carpuat, M., de Marneffe, M.C., Meza Ruiz, I.V. (eds.) Findings of the Association for Computational Linguistics: NAACL 2022. pp. 724–736. Association for Computational Linguistics, Seattle, United States (Jul 2022)

[15] Hathaliya, J.J., Tanwar, S., Sharma, P.: Adversarial learning techniques for security and privacy preservation: A comprehensive review. Security and Privacy **5**(3), e209 (2022)

[16] Iyyer, M., Wieting, J., Gimpel, K., Zettlemoyer, L.: Adversarial example generation with syntactically controlled paraphrase networks. In: Walker, M., Ji, H., Stent, A. (eds.) Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers). pp. 1875–1885. Association for Computational Linguistics, New Orleans, Louisiana (Jun 2018)

[17] Jia, R., Liang, P.: Adversarial examples for evaluating reading comprehension systems. In: Palmer, M., Hwa, R., Riedel, S. (eds.) Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing. pp. 2021–2031. Association for Computational Linguistics, Copenhagen, Denmark (Sep 2017)

[18] Jin, D., Jin, Z., Zhou, J.T., Szolovits, P.: Is bert really robust? a strong baseline for natural language attack on text classification and entailment. Proceedings of the AAAI Conference on Artificial Intelligence **34**(05), 8018–8025 (Apr 2020)

[19] Jin, D., Pan, E., Oufattole, N., Weng, W.H., Fang, H., Szolovits, P.: What disease does this patient have? a large-scale open domain question answering dataset from medical exams. Applied Sciences **11**(14), 6421 (2021)

[20] Kann, K., Rothe, S., Filippova, K.: Sentence-level fluency evaluation: References help, but can be spared! In: Korhonen, A., Titov, I. (eds.) Proceedings of the 22nd Conference on Computational Natural Language Learning. pp. 313–323. Association for Computational Linguistics, Brussels, Belgium (Oct 2018)

[21] Khashabi, D., Min, S., Khot, T., Sabharwal, A., Tafjord, O., Clark, P., Hajishirzi, H.: UNIFIEDQA: Crossing format boundaries with a single QA system. In: Cohn, T., He, Y., Liu, Y. (eds.) Findings of the Association for Computational Linguistics: EMNLP 2020. pp. 1896–1907. Association for Computational Linguistics, Online (Nov 2020)

[22] Klein, T., Nabi, M.: Learning to answer by learning to ask: Getting the best of gpt-2 and bert worlds. arXiv preprint arXiv:1911.02365 (2019)

[23] Klopfenstein, L.C., Delpriori, S., Malatini, S., Bogliolo, A.: The rise of bots: A survey of conversational interfaces, patterns, and paradigms. In: Proceedings of the 2017 Conference on Designing Interactive Systems. p. 555–565. DIS '17, Association for Computing Machinery, New York, NY, USA (2017)

[24] Ko, M., Lee, J., Kim, H., Kim, G., Kang, J.: Look at the first sentence: Position bias in question answering. In: Webber, B., Cohn, T., He, Y., Liu, Y. (eds.) Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP). pp. 1109–1121. Association for Computational Linguistics, Online (Nov 2020)

[25] Kočiský, T., Schwarz, J., Blunsom, P., Dyer, C., Hermann, K.M., Melis, G., Grefenstette, E.: The NarrativeQA reading comprehension challenge. Transactions of the Association for Computational Linguistics **6**, 317–328 (2018)

[26] Lan, Z., Chen, M., Goodman, S., Gimpel, K., Sharma, P., Soricut, R.: Albert: A lite bert for self-supervised learning of language representations. In: International Conference on Learning Representations (2020)

[27] Lewis, M., Liu, Y., Goyal, N., Ghazvininejad, M., Mohamed, A., Levy, O., Stoyanov, V., Zettlemoyer, L.: BART: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. In: Jurafsky, D., Chai, J., Schluter, N., Tetreault, J. (eds.) Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics. pp. 7871–7880. Association for Computational Linguistics, Online (Jul 2020)

[28] Li, D., Zhang, Y., Peng, H., Chen, L., Brockett, C., Sun, M.T., Dolan, B.: Contextualized perturbation for textual adversarial attack. In: Toutanova, K., Rumshisky, A., Zettlemoyer, L., Hakkani-Tur, D., Beltagy, I., Bethard, S., Cotterell, R., Chakraborty, T., Zhou, Y. (eds.) Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies. pp. 5053–5069. Association for Computational Linguistics, Online (Jun 2021)

[29] Li, J., Ji, S., Du, T., Li, B., Wang, T.: Textbugger: Generating adversarial text against real-world applications. In: Proceedings 2019 Network and Distributed System Security Symposium. NDSS 2019, Internet Society (2019)

[30] Li, J., Liu, W.: Summarization attack via paraphrasing (student abstract). In: Proceedings of the AAAI Conference on Artificial Intelligence. vol. 37, pp. 16250–16251 (2023)

[31] Li, L., Ma, R., Guo, Q., Xue, X., Qiu, X.: BERT-ATTACK: Adversarial attack against BERT using BERT. In: Webber, B., Cohn, T., He, Y., Liu, Y. (eds.) Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP). pp. 6193–6202. Association for Computational Linguistics, Online (Nov 2020)

[32] Liang, B., Li, H., Su, M., Bian, P., Li, X., Shi, W.: Deep text classification can be fooled. In: Proceedings of the Twenty-Seventh International Joint Conference on Artificial Intelligence. p. 4208–4215. International Joint Conferences on Artificial Intelligence Organization (Jul 2018)

[33] Lin, C.Y.: Rouge: A package for automatic evaluation of summaries. In: Text summarization branches out. pp. 74–81 (2004)

[34] Luong, T., Pham, H., Manning, C.D.: Effective approaches to attention-based neural machine translation. In: Màrquez, L., Callison-Burch, C., Su, J. (eds.) Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing. pp. 1412–1421. Association for Computational Linguistics, Lisbon,

Portugal (Sep 2015)

[35] Martinez-Gil, J.: A survey on legal question–answering systems. Computer Science Review **48**, 100552 (2023)

[36] Mikolov, T., Chen, K., Corrado, G., Dean, J.: Efficient estimation of word representations in vector space. In: Proceedings of the 1st International Conference on Learning Representations (ICLR) (2013)

[37] Miller, G.A.: WordNet: A lexical database for English. In: Speech and Natural Language: Proceedings of a Workshop Held at Harriman, New York, February 23-26, 1992 (1992)

[38] Mozes, M., Stenetorp, P., Kleinberg, B., Griffin, L.: Frequency-guided word substitutions for detecting textual adversarial examples. In: Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume. pp. 171–186. Association for Computational Linguistics, Online (Apr 2021)

[39] Naber, D.: A Rule-Based Style and Grammar Checker. GRIN Verlag (2003)

[40] Nuruzzaman, M., Hussain, O.K.: A survey on chatbot implementation in customer service industry through deep neural networks. In: 2018 IEEE 15th International Conference on e-Business Engineering (ICEBE). pp. 54–61 (2018)

[41] Papineni, K., Roukos, S., Ward, T., Zhu, W.J.: Bleu: a method for automatic evaluation of machine translation. In: Proceedings of the 40th annual meeting of the Association for Computational Linguistics. pp. 311–318 (2002)

[42] Radford, A., Wu, J., Child, R., Luan, D., Amodei, D., Sutskever, I., et al.: Language models are unsupervised multitask learners. OpenAI blog **1**(8), 9 (2019)

[43] Raffel, C., Shazeer, N., Roberts, A., Lee, K., Narang, S., Matena, M., Zhou, Y., Li, W., Liu, P.J.: Exploring the limits of transfer learning with a unified text-to-text transformer. Journal of machine learning research **21**(140), 1–67 (2020)

[44] Rajpurkar, P., Jia, R., Liang, P.: Know what you don't know: Unanswerable questions for SQuAD. In: Gurevych, I., Miyao, Y. (eds.) Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers). pp. 784–789. Association for Computational Linguistics, Melbourne, Australia (Jul 2018)

[45] Rajpurkar, P., Zhang, J., Lopyrev, K., Liang, P.: SQuAD: 100,000+ questions for machine comprehension of text. In: Su, J., Duh, K., Carreras, X. (eds.) Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing. pp. 2383–2392. Association for Computational Linguistics, Austin, Texas (Nov 2016)

[46] Ren, S., Deng, Y., He, K., Che, W.: Generating natural language adversarial examples through probability weighted word saliency. In: Korhonen, A., Traum, D., Màrquez, L. (eds.) Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics. pp. 1085–1097. Association for Computational Linguistics, Florence, Italy (Jul 2019)

[47] Rosenthal, S., Bornea, M., Sil, A.: Are multilingual bert models robust? a case study on adversarial attacks for multilingual question answering. arXiv preprint arXiv:2104.07646 (2021)

[48] Sanh, V., Debut, L., Chaumond, J., Wolf, T.: DistilBERT, a distilled version of BERT: smaller, faster, cheaper and lighter. In: Proceedings of the 5th Workshop on Energy Efficient Machine Learning and Cognitive Computing. NeurIPS, Vancouver, Canada (2019)

[49] Shinoda, K., Sugawara, S., Aizawa, A.: Penalizing confident predictions on largely perturbed inputs does not improve out-of-distribution generalization in question answering. In: Proceedings of the Workshop on Knowledge Augmented Methods for NLP (KnowledgeNLP) at AAAI 2023 (2023)

[50] Soares, M.A.C., Parreiras, F.S.: A literature review on question answering techniques, paradigms and systems. Journal of King Saud University-Computer and Information Sciences **32**(6), 635–646 (2020)

[51] Stiennon, N., Ouyang, L., Wu, J., Ziegler, D., Lowe, R., Voss, C., Radford, A., Amodei, D., Christiano, P.F.: Learning to summarize with human feedback. Advances in Neural Information Processing Systems **33**, 3008–3021 (2020)

[52] Sun, H., Zhu, T., Zhang, Z., Jin, D., Xiong, P., Zhou, W.: Adversarial attacks against deep generative models on data: a survey. IEEE Transactions on Knowledge and Data Engineering **35**(4), 3367–3388 (2021)

[53] Talmor, A., Berant, J.: MultiQA: An empirical investigation of generalization and transfer in reading comprehension. In: Korhonen, A., Traum, D., Màrquez, L. (eds.) Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics. pp. 4911–4921. Association for Computational Linguistics, Florence, Italy (Jul 2019)

[54] Toutanova, K., Klein, D., Manning, C.D., Singer, Y.: Feature-rich part-of-speech tagging with a cyclic dependency network. In: Proceedings of the 2003 Conference of the North American Chapter of the Association for Computational Linguistics on Human Language Technology - Volume 1. p. 173–180. NAACL '03, Association for Computational Linguistics, USA (2003)

[55] Trischler, A., Wang, T., Yuan, X., Harris, J., Sordoni, A., Bachman, P., Suleman, K.: NewsQA: A machine comprehension dataset. In: Blunsom, P., Bordes, A., Cho, K., Cohen, S., Dyer, C., Grefenstette, E., Hermann, K.M., Rimell, L., Weston, J., Yih, S. (eds.) Proceedings of the 2nd Workshop on Representation Learning for NLP. pp. 191–200. Association for Computational Linguistics, Vancouver, Canada (Aug 2017)

[56] Wallace, E., Feng, S., Kandpal, N., Gardner, M., Singh, S.: Universal adversarial triggers for attacking and analyzing NLP. In: Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP). pp. 2153–2162. Association for Computational Linguistics, Hong Kong, China (November 2019)

[57] Wallace, E., Rodriguez, P., Feng, S., Yamada, I., Boyd-Graber, J.: Trick me if you can: Human-in-the-loop generation of adversarial examples for question answering. Transactions of the Association for Computational Linguistics **7**, 387–401 (2019)

[58] Wang, B., Pei, H., Pan, B., Chen, Q., Wang, S., Li, B.: T3: Tree-autoencoder constrained adversarial text generation for targeted attack. In: Webber, B., Cohn,

T., He, Y., Liu, Y. (eds.) Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP). pp. 6134–6150. Association for Computational Linguistics, Online (Nov 2020)

[59] Wang, Z.: Modern question answering datasets and benchmarks: A survey. arXiv preprint arXiv:2206.15030 (2022)

[60] Xiao, T., Zhu, J.: Introduction to transformers: an nlp perspective (2023)

[61] Xu, K., Ba, J., Kiros, R., Cho, K., Courville, A., Salakhudinov, R., Zemel, R., Bengio, Y.: Show, attend and tell: Neural image caption generation with visual attention. In: International conference on machine learning. pp. 2048–2057. PMLR (2015)

[62] Yang, X., Liu, W., Zhang, S., Liu, W., Tao, D.: Targeted attention attack on deep learning models in road sign recognition. IEEE Internet of Things Journal **8**(6), 4980–4990 (2020)

[63] Yasunaga, M., Kasai, J., Radev, D.: Robust multilingual part-of-speech tagging via adversarial training. In: Walker, M., Ji, H., Stent, A. (eds.) Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers). pp. 976–986. Association for Computational Linguistics, New Orleans, Louisiana (Jun 2018)

[64] Yigit, G., Amasyali, M.F.: From text to multimodal: A comprehensive survey of adversarial example generation in question answering systems. Knowledge and Information Systems **66**, 7165–7204 (2024)

[65] Yu, J., Liu, W., Qiu, S., Su, Q., Wang, K., Quan, X., Yin, J.: Low-resource generation of multi-hop reasoning questions. In: Jurafsky, D., Chai, J., Schluter, N., Tetreault, J. (eds.) Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics. pp. 6729–6739. Association for Computational Linguistics, Online (Jul 2020)

[66] Zang, Y., Qi, F., Yang, C., Liu, Z., Zhang, M., Liu, Q., Sun, M.: Word-level textual adversarial attacking as combinatorial optimization. In: Jurafsky, D., Chai, J., Schluter, N., Tetreault, J. (eds.) Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics. pp. 6066–6080. Association for Computational Linguistics, Online (Jul 2020)

[67] Zeng, J., Zheng, X., Xu, J., Li, L., Yuan, L., Huang, X.: Certified robustness to text adversarial attacks by randomized [mask]. Computational Linguistics **49**(2), 395–427 (jun 2023)

[68] Zhang, J., Zhao, Y., Saleh, M., Liu, P.: Pegasus: Pre-training with extracted gap-sentences for abstractive summarization. In: International conference on machine learning. pp. 11328–11339. PMLR (2020)

[69] Zhu, F., Lei, W., Wang, C., Zheng, J., Poria, S., Chua, T.S.: Retrieving and reading: A comprehensive survey on open-domain question answering. arXiv preprint arXiv:2101.00774 (2021)

[70] Zhuang, L., Wayne, L., Ya, S., Jun, Z.: A robustly optimized BERT pre-training approach with post-training. In: Li, S., Sun, M., Liu, Y., Wu, H., Liu, K., Che, W., He, S., Rao, G. (eds.) Proceedings of the 20th Chinese National Conference on Computational Linguistics. pp. 1218–1227. Chinese Information Processing

Society of China, Huhhot, China (Aug 2021)