Evaluating and Enhancing the Robustness of Retrieval-based Dialogue Systems with Adversarial Examples

Jia Li[1], Chongyang Tao[1], Nanyun Peng[2], Wei Wu[3], Dongyan Zhao[1], and Rui Yan[1⋆]

[1] Institute of Computer Science and Technology, Peking University, Beijing, China
{lijiaa,chongyangtao,ruiyan,zhaody}@pku.edu.cn
[2] Information Sciences Institute, University of Southern California
npeng@isi.edu
[3] Microsoft Corporation, Beijing
wuwei@microsoft.com

Abstract. Retrieval-based dialogue systems have shown strong performances on both consistency and fluency according to several recent studies. However, their robustness towards malicious attacks remains largely untested. In this paper, we generate adversarial examples in black-box settings to evaluate the robustness of retrieval-based dialogue systems. On three representative retrieval-based dialogue models, our attacks reduce $R_{10}@1$ by 38.3%, 45.0% and 31.5% respectively on the Ubuntu dataset. Moreover, with adversarial training using our generated adversarial examples, we significantly improve the robustness of retrieval-based dialogue systems. We conduct thorough analysis to understand the robustness of retrieval-based dialog systems. Our results provide new insights to facilitate future work on building more robust dialogue systems.

Keywords: Retrieval-based Dialogue Systems · Adversarial Examples.

1 Introduction

Intelligent agents that communicate with human in natural language have been applied to many down-stream applications, such as question-answering, negotiation, electronic commerce [17, 18, 6]. Specially, retrieval-based dialogue systems have shown strong performances on both consistency and fluency. The current state-of-the-art system achieves 78.6% $R_{10}@1$ on the Ubuntu dataset. However, achieving excellent performance does not indicate that retrieval-based dialogue systems really understand natural language and will also work well when countering malicious attacks. Currently, retrieval-based dialogue systems use a test set to measure models. High accuracy on test set indicates an excellent model on condition that the test set represents the real-world [15]. However, since the test set is usually created along with a training set, the test set is likely to have the same distribution as its corresponding training set, which does not necessarily represent real-world scenarios.

⋆ Corresponding author. (Email: ruiyan@pku.edu.cn)

Table 1. An example from the Douban dataset. The model labels the original positive response correctly with Label 1 (in blue), but the model is fooled by our adversarial example generated by replacing words with synonyms (RSW) (in red).

Context	
Speaker A	When others accept me, I always deny myself in public unconsciously.
Speaker B	Confidence is an inner emotion.
Speaker A	How to cultivate confidence?
Speaker B	You must depend on yourself.
Speaker A	I am excellent.
$\text{Response}_{\text{ori}}$	Am I excellent ? (Label=1)
$\text{Response}_{\text{RSW}}$	Am I outstanding ? (Label=0)

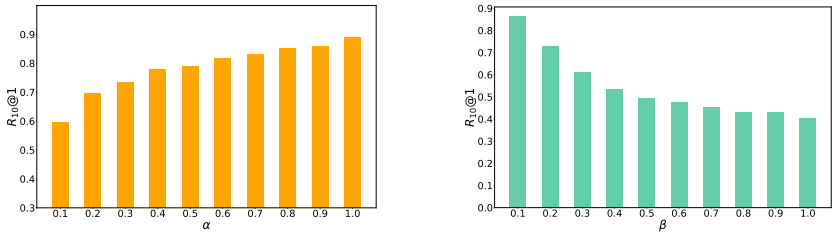
To better understand the robustness of retrieval-based dialogue systems, we conduct empirical studies and propose methods to generate adversarial examples to attack retrieval-based dialogue systems in black-box settings. There are several interesting findings. First, we observe that the performance of models is related to the degree of word overlap between context and response. High accuracy corresponds to high word overlap. Besides, we consider adversarial attacks by inserting important words using TF-IDF score, synonym substitution, shuffling words, and repeating some words to generate adversarial examples in the test set. We also generate uninformative and generic responses to evaluate the sensitivity of the retrieval-based models to generic responses. Table 1 gives an example of one adversarial example we generated by replacing words with synonyms.

To improve the robustness of retrieval-base dialogue systems, we conduct adversarial training [12] to protect the retrieval-based dialogue models from attacks. Specifically, we randomly select $100,000$ examples from the training set to generate adversarial examples and train the models again.

Challenges and our contributions. Earlier adversarial example studies focused on image classification. Several recent works have extended it to natural language processing (NLP) tasks, such as text classification [4], question-answering [15], and reading comprehension [8]. Different from these tasks, the evaluation of retrieval-based dialogue systems has unique characteristics. Specifically, if we change positive examples into negative responses, the accuracy of the models can not be evaluated by standard evaluation metrics used in the previous works [19, 18]. This leads to several challenges including 1) How to measure the success of an attack and 2) how to make effective adversarial examples considering the characteristics of retrieval-based dialogue models and the data sets.

In this paper, we tackle the aforementioned challenges by proposing new evaluation metrics and carefully designing adversarial examples. We highlight our major contributions as following:

- To the best of our knowledge, this is the first work to measure the robustness of retrieval-based dialogue systems under adversarial attacks.
- Carefully design adversarial example generation methods, which successfully fool retrieval-based dialog systems.



(a) (b)

Fig. 1. (a) The effect of α on the performance of SMN on the Ubuntu data. (b) The effect of β on the performance of SMN on the Ubuntu data. In both figures, we report the most important metric $R_{10}@1$.

– We significantly improve the robustness of retrieval-based dialogue systems with our proposed adversarial training methods.

2 Empirical Observations

In this section, we present two empirical observations about the performance of matching model. All experimental results in this section are based on SMN [19] (the most representative model) with Ubuntu Dialogue Corpus [11], which is the most typical data set for retrieval-based dialogue systems.

1) *The performance of context-response matching models is closely related to the degree of word overlap between context and its corresponding response.*

Suppose that l_c represents the set of words contained in context c and l_{r^+} is the set of words contained in positive response r^+. $|l_c \cap l_{r^+}|$ represents word overlap numbers between context c and positive response r^+. Then we obtain normalized word overlap degree α, which can be formulated as:

$$\alpha = \frac{|l_c \cap l_{r^+}|}{min(|l_c|, |l_{r^+}|)}. \tag{1}$$

Besides, we also take into account the word overlap between context and its corresponding negative responses. Suppose that $\mathcal{R}^-$ represents the negative responses pool for each context in test set. $l_{r_j^-}$ is the set of words contained in negative response r_j^- among $\mathcal{R}^-$ and $|l_c \cap l_{r_j^-}|$ represents word overlap numbers between context c and negative response r_j^-. β is defined as:

$$\beta = \frac{\arg\max_{r_j^- \in \mathcal{R}^-}(|l_c \cap l_{r_j^-}|)}{|l_c \cap l_{r^+}|}. \tag{2}$$

Figure 1 shows that how the performance of SMN model changes along with word overlap degree α and β. From Figure 1(a), we can see that the performance of mod-

els gradually improves with the increase of word overlap degree α. According to Figure 1(b), we observe that the higher word overlap numbers between contexts and negative responses is, the more models would be interfered in choosing a positive response as a correct answer, which leads to lower accuracy. We suspect that a critical factor for the performance of models is the degree of word overlap between contexts and responses.

2) *Over-stability: The performance of context-response matching models mostly depends on a few important words in the response.*

To determine the words set in positive responses that network considers most important, we compute the importance of each word in positive responses. We calculate the relative change of $R_{10}@1$ when a particular word is erased.

Let $D = \{(c_i, \{r_{i,j}^+\}_{j=1}^{n_i^+}, \{r_{i,k}^-\}_{k=1}^{n_i^-})\}_{i=1}^N$ is a test set, where c_i is a conversation context, $\forall j \in \{1, \ldots, n_i^+\}$, $r_{i,j}^+$ is a positive response candidate that properly replies to c_i, and $\forall k \in \{1, \ldots, n_i^-\}$, $r_{i,k}^-$ is a negative response candidate. $g(c_i, r_{i,j}^+)$ denotes a score of correct label between positive response $r_{i,j}^+$ and its corresponding context c_i. For each positive example in the test set, we erase one word w_{ijz} in positive response at a time, then compute relative change of the score. The importance $s(w_{ijz})$ of word w_{ijz} could be formulated as:

$$s(w_{ijz}) = \frac{1}{|N|} \sum_{n \in N} \frac{g(c_i, r_{i,j}^+) - g(c_i, r_{i,j}^+ - w_{ijz})}{g(c_i, r_{i,j}^+)}, \tag{3}$$

where $g(c_i, r_{i,j}^+ - w_{ijz})$ represents the score between the rest part of a positive response $r_{i,j}^+$ removing z-th word w_{ijz} and its corresponding context c_i. N is the word w_{ijz} appearing times in positive response $r_{i,j}^+$. A high score $s(w_{ijz})$ represents that word w_{ijz} has a high attribution.

We calculate the $R_{10}@1$ score on context-response matching models when we only keep top $k \in (1, 2, 3, 4, 5, 6)$ important words in responses. Figure 2 shows how the relative accuracy changes when k varies from 1 to 6, in which relative changes represents that the current accuracy is divided by original accuracy. The result shows that remaining one word in the response enables the model to achieve more than 70% relative accuracy. The relative accuracy increases almost monotonically with the number of reserved words in responses. This indicates another critical factor that models select a response depending on a set of important words.

3 Attacking Approaches

In this paper, we consider a series of adversarial approaches to evaluate the robustness of existing context-response matching models.

3.1 Insert Important Words

The basic idea of the algorithm TF-IDF is to represent a context c_i as a vector $\boldsymbol{c}_i = (c_{i1}, c_{i2}, ..., c_{ik})$, where k is the number of words in the context. Suppose that w_{im} is

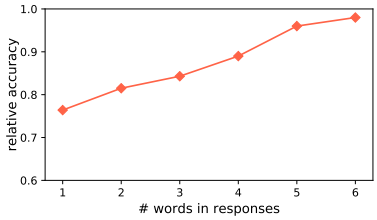


Fig. 2. Relative accuracy as the number of reserved words changes on the Ubuntu data, compared with its original accuracy.

the m-th word of i-th context c_i, $TF(w_{im})$ is the number of times w_{im} occurs in all contexts. $DF(w_{im})$ is the number of contexts in which the word w_{im} occurs at least one time. $IDF(w_{im})$ can be formulated as:

$$IDF(w_{ij}) = \log(\frac{|M|}{DF(w_{ij})}), \tag{4}$$

where M is the number of all contexts. $IDF(w_{im})$ is low if the word w_{im} is in many contexts and is high if the word w_{im} occurs less times [9]. The feature value c_{im} of word w_{im} can be formulated as:

$$c_{im} = TF(w_{im}) \cdot IDF(w_{im}) \tag{5}$$

Then we get the weight of word w_{im} in the context. For each negative examples, we select the top three words (c_i^1, c_i^2, c_i^3) with high TF-IDF values in context c_i to replace three words with the same part of speech in its corresponding negative response r_i , where $r_i = (r_i^1, ..., r_i^i, ..., r_i^j, ..., r_i^k, ..., r_i^q)$. The adversarial negative response can be formulated as $a_i = (r_i^1, ..., c_i^1, ..., c_i^2, ..., c_i^3, ..., r_i^q)$. Table 2 shows an example about this attack. We identify the part of speech (POS) of words by using the POS tagger in the NLTK library for the Ubuntu data and jieba for the Douban data.

3.2 Replace Words by Synonyms

We conduct synonym replacement (excluding stops words and named entities) utilizing WordNet from NLTK. The negative responses and contexts are unaltered. This adversarial method keeps the syntax, semantics and meaning of positive responses invariant, which will not affect the performance of models ideally.

3.3 Shuffling Words

To learn if models could understand words order in sentences, we randomly shuffle words in positive responses. Negative responses and contexts keep unchanged. Since words order alters, positive responses turn into negative responses. In the adversarial attack, the lower the adversarial evaluation metrics are, the more robust the models are.

Table 2. An example from the Douban data. The model labels an original negative response correctly with Label 0 (in blue), but is fooled by inserting important words (IIW) (in red).

Context	
Speaker A	What browser do you use?
Speaker B	Which one do you think best?
Speaker A	QQ browser.
Speaker B	Good eye.
Speaker A	Thank you.
Speaker B	Your avatar is stupid.
$\text{Response}_{\text{ori}}$	That is good. I draw abstractionism. (Label=0)
$\text{Response}_{\text{IIW}}$	That is stupid. I use browser. (Label = 1)

3.4 Repeat Some Words

It is well known that since human labeling is expensive and exhausting, most of the existing works adopt a simple method to automatically build a data set, in which response candidates are almost obtained from generated-based models on most practical applications. However, the generated-based models often generate responses with duplicate words. It is necessary to verify the robustness of models in this case.

Specifically, we consider two strategies to repeat words in positive responses, namely $\text{RSW}_{\frac{L}{2}}$ and RSW_1. In the first strategy, we randomly choose $\frac{L}{2}$ words to repeat one time in a positive response, where L is the length of the positive response. In the other strategy, we randomly select one word to repeat $\frac{L}{2}$ times in the positive response. Considering changes in sentence fluency, the positive responses become negative responses. The model should be able to distinguish the unnatural behavior and adversarial evaluation metrics should decrease ideally.

3.5 Retain the Nouns, Pronouns and Verbs

In this attack method, we observe that whether models could recognize integrity of sentence components. Using the POS tagger functionality of NLTK library, the positive responses only contain nouns, pronouns and verbs by removing adjectives, adverbs and prepositions, etc. This attack method makes positive responses lose their original meaning. Hence, the positive responses turn into negative responses due to incomplete sentence components.

3.6 Neutral and Generic Responses

Neutral and generic responses are readily regarded as suitable responses in most cases, but these responses contain little information and are meaningless. To understand whether models could distinguish neutral and generic responses in the vector space correctly, we come up with some neutral and generic responses, such as “I am sorry can you repeat” and “Fantastic that sounds good”. We replace all positive responses with neutral responses in the test set to evaluate the robustness of models for neutral responses.

Table 3. Results of IIW, RSW adversarial evaluation on the SMN, DAM, MFRN models. All three models can be fooled by adversarial examples. "BASE" represents the baseline of models.

	Attack	Ubuntu Corpus				Douban Conversation Corpus					
		$\mathbf{R_2@1}$	$\mathbf{R_{10}@1}$	$\mathbf{R_{10}@2}$	$\mathbf{R_{10}@5}$	**MAP**	**MRR**	**P@1**	$\mathbf{R_{10}@1}$	$\mathbf{R_{10}@2}$	$\mathbf{R_{10}@5}$
SMN	BASE	0.926	0.726	0.847	0.961	0.529	0.569	0.397	0.233	0.396	0.724
	IIW	0.625	0.345	0.443	0.622	0.389	0.426	0.231	0.123	0.233	0.526
	RSW	0.621	0.342	0.436	0.621	0.374	0.395	0.165	0.087	0.225	0.473
DAM	BASE	0.938	0.767	0.874	0.969	0.550	0.601	0.427	0.254	0.410	0.757
	IIW	0.629	0.358	0.447	0.624	0.368	0.403	0.225	0.113	0.198	0.493
	RSW	0.637	0.366	0.461	0.632	0.382	0.407	0.193	0.109	0.215	0.484
MFRN	BASE	0.945	0.786	0.886	0.976	0.571	0.617	0.448	0.276	0.435	0.783
	IIW	0.642	0.382	0.471	0.642	0.386	0.422	0.244	0.134	0.232	0.498
	RSW	0.853	0.569	0.722	0.897	0.388	0.419	0.224	0.128	0.242	0.497

4 Adversarial Training

Adversarial training is becoming more and more popular to improve the robustness of machine learning models [14, 12, 3]. We train retrieval-based dialogue models using adversarial examples and observe whether these models can become more robust.

In standard adversarial training for neural networks models [7, 8], adversarial examples for adversarial training are produced through the same attack methods in the test set. We also perform adversarial training with the attack methods mentioned above in this paper. Firstly, we randomly select 100,000 examples to form a $\mathcal{F}^-$ from the training set. In the case of inserting important words attack, we randomly select words in each context in the pool $\mathcal{F}^-$ and use them to replace words in its corresponding negative response with same part of speech. The number of words being replaced is one-ninth of the length of the context. To the synonyms substitution attack, we replace words with their synonyms in positive responses in the pool $\mathcal{F}^-$. For the rest of attack methods, we change positive examples in the pool $\mathcal{F}^-$ into negative examples according to the above attack methods separately, and insert them into training data. Meanwhile, we reserve original positive examples in the training set.

5 Experiments

We conduct comprehensive experiments to evaluate and analyze the robustness of three representative multi-turn response selection models with different levels of complexity, namely SMN [19], DAM [22] and MFRN [18]. Moreover, the robustness of these models can be significantly improved by adversarial training. We denote attack methods with inserting important words, replacing words by synonyms, shuffling words, retaining the Nouns, pronouns and verbs, and neutral and generic responses as IIW, RSW, SOW, RNPV and NGR respectively.

5.1 Experimental Setup

We conduct experiments on two public data sets, including Ubuntu Dialogue Corpus [11] collected from chat logs of the Ubuntu Forum and Douban Conversation Corpus

Table 4. Results of SOW, RLW, RNPV and NGR adversarial evaluation on the SMN, DAM, MFRN models. All three mo dels can be fooled by adversarial examples.

		Ubuntu Corpus				Douban Conversation Corpus					
		A_2@1	**A_{10}@1**	**A_{10}@2**	**A_{10}@5**	**AAP**	**ARR**	**A@1**	**A_{10}@1**	**A_{10}@2**	**AR_{10}@5**
SMN	SOW	0.917	0.711	0.832	0.953	0.533	0.571	0.388	0.227	0.401	0.756
	RLW $_{\frac{L}{2}}$	0.908	0.695	0.820	0.941	0.524	0.567	0.379	0.221	0.393	0.733
	RLW_1	0.903	0.698	0.812	0.938	0.527	0.563	0.391	0.223	0.396	0.751
	RNPV	0.907	0.685	0.827	0.932	0.505	0.549	0.355	0.207	0.368	0.746
	NGR	0.894	0.589	0.807	0.926	0.213	0.212	0.076	0.031	0.062	0.147
DAM	SOW	0.933	0.765	0.866	0.962	0.548	0.587	0.426	0.250	0.410	0.758
	RLW $_{\frac{L}{2}}$	0.929	0.734	0.837	0.947	0.539	0.587	0.406	0.239	0.400	0.769
	RLW_1	0.935	0.736	0.836	0.952	0.544	0.594	0.423	0.250	0.401	0.758
	RNPV	0.921	0.726	0.832	0.947	0.540	0.581	0.396	0.241	0.397	0.781
	NGR	0.907	0.567	0.814	0.936	0.349	0.297	0.094	0.065	0.132	0.346
MFRN	SOW	0.942	0.778	0.884	0.979	0.556	0.603	0.439	0.258	0.409	0.783
	RLW $_{\frac{L}{2}}$	0.931	0.763	0.857	0.975	0.551	0.602	0.437	0.262	0.413	0.758
	RLW_1	0.934	0.767	0.859	0.963	0.552	0.595	0.426	0.259	0.411	0.762
	RNPV	0.921	0.754	0.832	0.951	0.522	0.568	0.409	0.235	0.388	0.719
	NGR	0.873	0.417	0.711	0.773	0.357	0.304	0.107	0.079	0.154	0.378

[19] collected from Douban group[4]. In the both data sets, we limit the maximun number of utterances in each context as 10 and the maximun number of words in each utterance as 50 for computational efficiency. We perform zero-padding or truncation when necessary. Word embedding is pre-trained with Word2Vec [13] on the training sets of both Ubuntu and Douban, and the dimension of word vectors is 200. In the adversarial evaluation metrics, we label the examples that are attacked by shuffling words, repeating some words, retaining only some words, and neutral responses as 1, though they are not really positive examples in the test set. For adversarial training, the examples disrupted by the attack methods mentioned above are labeled as 0 in the training set. For IIW and RSW attacks, the labels of examples remain unchanged in both training set and test set.

5.2 Evaluation Metrics

Following the setting of previous works [19, 22, 18], we employ $R_n@k$ for both Ubuntu and douban datasets, and employ mean average precision (MAP), mean reciprocal rank (MRR) and precision-at-one (P@1) for Douban dataset.

Adversarial Evaluation Metrics. For each context in the test set, 10 response candidates are retrieved from an index and are divided into positive responses r_p with label 1 and negative responses r_n with label 0, according to their appropriateness regarding to the context. In this paper, our attack methods would disturb a positive response r_p into a new-negative-response r_a. Since 10 response candidates are all negative responses for each context, standard evaluation metrics are no longer valid. To evaluate the robustness of retrieval-based dialogue systems, we proposed adversarial evaluation metrics ($A_n@k$, AAP, ARR, A@1), indicating the degree of success of an attack method. The value is in region of $[0, 1]$, and a larger value indicates a more successful attack. $A_n@k$ is

[4] https://www.douban.com/group

Table 5. Results of adversarial training. Models are trained on IIW, RSW attacking training set and test on original test set and IIW, RSW attacking test set. "Ori" represents the original training set or test set.

	SMN			DAM			MRFN		
	ORI	IIW	RSW	Ori	IIW	RSW	Ori	IIW	RSW
Ori	0.726	0.737	0.729	0.767	0.768	0.768	0.786	0.791	0.789
IIW	0.345	0.458	-	0.358	0.486	-	0.382	0.525	-
RSW	0.342	-	0.406	0.366	-	0.434	0.529	-	0.596

defined as the recall of a new-negative-response r_a among the k selected best-matched response from n available candidates. Similar to $\mathrm{A}_n@k$, the rest of adversarial evaluation metrics (AAP, ARR and A@1) are calculated in the same way, except that the positive response r_p is replaced by a new-negative-response r_a. Note that this is the first work on attacking retrieval-based dialogue systems, so there is no previous results that could be included to compare with.

5.3 Adversarial Attack Results and Analysis

Table 3 and Table 4 report the performance of retrieval-based dialogue models on our proposed attack methods. We can see that each attack method leads to a significant decrease in the standard evaluation metrics, while obtains remarkable high values on adversarial evaluation metrics.

Insert Important Words. In the Table 3, we can observe that the result $\mathrm{R}_{10}@1$ drops by 38.1%, 40.9% and 40.4% against IIW attack on three models on Ubuntu data. Meanwhile, on Douban data, the performance P@1 decreases by nearly 50% on all models.

Replace Words by Synonyms. In this attack, we only replace words by synonyms in positive responses, which does not change the meaning of samples. The detailed results are shown in Table 3. We can observe that the adversarial examples can achieve a successful attack. The main reason for poor performance might be that synonyms substitution leads to lower word overlap. Furthermore, we have included an adversarial example played by RSW in Table 1. From this example, we can see that RSW attack can generate adversarial responses with unchanged meaning which could fool models to make terrible selection.

Shuffle Words. All the positive examples are changed into negative examples after SOW attack in the test set. To solve this problem , we use adversarial evaluation metrics to test the robustness of models. From Table 4, we can see that SOW attack can achieve $\mathrm{A}_{10}@1$ 71.1% and 77.8% scores on SMN and MFRN respectively, being similar to $\mathrm{R}_{10}@1$ values, which reveals that the models can still choose scrambled responses as positive responses. Hence, we can conclude that the context-response matching models do not really understand words order in sentences.

Repeat Some Words. To determine the influence of an unnatural variant from positive responses to negative responses, we repeat some words in positive responses. From Table 4, RLW attack achieves high $A_{10}@1$ scores, which indicates that networks could hardly distinguish unnatural sentences.

Table 6. Results of adversarial training. Models are trained on SOW, RSW and RNPV attacking training set and test on original test set and SOW, RSW and RNPV attacking test set. "Ori" represents the original training set or test set.

	SMN				DAM				MRFN			
	SOW	$RSW_{L/2}$	RSW_1	RNPV	SOW	$RSW_{L/2}$	RSW_1	RNPV	SOW	$RSW_{L/2}$	RSW_1	RNPV
Ori	0.723	0.728	0.741	0.725	0.756	0.763	0.759	0.763	0.785	0.790	0.789	0.786
SOW	0.057	-	-	-	0.124	-	-	-	0.058	-	-	-
$RSW_{L/2}$	-	0.032	-	-	-	0.047	-	-	-	0.060	-	-
RSW_1	-	-	0.105	-	-	-	0.094	-	-	-	0.076	-
RNPV	-	-	-	0.047	-	-	-	0.058	-	-	-	0.042

Retain the Nouns, Pronouns and Verbs. Our intention is to check whether networks can judge integrity of sentence components. From the results in Table 4, although sentences are incomplete, we can observe that $A_{10}@1$ scores are 3.80% and 2.67% less than $R_{10}@1$ on average on two data. The results demonstrate that sentence components are not really understood by models.

Neutral and Generic Responses. Table 4 also shows the performance of models when neutral responses are used. In this work, we only conduct experiment on one neutral response —"I am sorry can you repeat", which could be applied in most situations. We can see that models could achieve 3.1%, 6.5% and 7.9% at $A_{10}@1$ on Douban data. Moreover, on Ubuntu data, $A_{10}@1$ is less than 60%. The results indicate that networks have ability to discriminate neutral and generic responses.

5.4 Adversarial Training Results

We train models on our adversarial examples and observe whether networks could learn to be more robust. The results are shown in Table 5 and Table 6. From Table 5, we observe that the adversarial trained models achieve much better performance against IIW attack—$R_{10}@1$ score increases by 11.3%, 12.8% and 14.3% respectively on Ubuntu data. For further investigation, we train models on IIW examples and test on original test set. The results demonstrate that training models on adversarial examples generated by IIW attack not only significantly improve the performance of models on IIW attacked test set, but also improve accuracy on original test set, although improvement is limited. RSW adversarial training has the same performance.

From Table 6, we can see that models achieve consistently better performance against SOW, $RLW_{\frac{L}{2}}$, RLW_1 and RNPV attacks. The $A_{10}@1$ scores drops by 67.1%, 70.2%, 63.8% and 67.2% on the four attacks on average. For instance, the performance of MFRN reduced from 76.3% to 0.6% by $RLW_{\frac{L}{2}}$ adversarial training. These results indicate that the recognition ability of models to word order, sentence naturalness and sentence component integrity is dramatically enhanced. Moreover, adversarial trained models have limited effects on the original text, which reflects our attack examples can effectively enhance networks to resist attacks without damaging experimental results on original test set.

6 Related Work

Generating adversarial examples to evaluate the robustness of models has been proposed in different NLP tasks. [16] utilize Fast Gradient Sign method to generate adversarial examples that solve discrete problems in text on RNN/LSTM models. [4] propose a white-box adversary to trick a character-level neural networks, based on the gradients of the one-hot input vectors. [8] test the SQuAD reading comprehension task by inserting adversarial sentences into paragraphs. [1] aim to fool sentiment analysis and textual entailment models by a black-box population-based optimization algorithm. [2] confirm that character-level neural machine models are sensitive with synthetic and natural sources of noise, such as keyboard typos. [10] get important words by erasing them in sentiment analysis task and locates those words by using reinforcement learning. [5] present DeepWordBug algorithm to generate small text perturbations in a black-box setting on deep learning classification task. [15] use integrated gradients to learn the attribution of words (important words) and attack models on question answering based on images, tables and passages. [20] propose a greedy algorithm to swap words and character, and utilize a Gumbel softmax function to reduce the computation. [21] use Generative Adversarial Networks to generating adversarial examples.

However, little attention has been paid to context-response matching models. To the best of our knowledge, we are the first to evaluate the robustness of retrieval-based dialogue systems. Moreover, we take advantage of unique features of matching models by our empirical observation.

7 Conclusions

We analyze models through empirical observation on word overlap and word attributions, which helps us identify the weakness of context-response matching models and attack models more effectively. We generate adversarial examples from the perspectives of word overlap, words order, sentence fluency and sentence component. Our experimental results indicate that the current context-response matching models are not robust in the face of malicious attacks. Furthermore, by adversarial training using our attack methods, we can significantly improve the robustness of the retrieval-based dialogue systems. We believe our work would aid the development of deep neural networks.

Acknowledgments

We thank the reviewers for their valuable comments. This work was supported by the National Key Research and Development Program of China (No. 2017YFC0804001), the National Science Foundation of China (NSFC No. 61876196, NSFC No. 61828302, and NSFC No. 61672058).

References

1. Alzantot, M., Sharma, Y., Elgohary, A., Ho, B.J., Srivastava, M., Chang, K.W.: Generating natural language adversarial examples. arXiv preprint arXiv:1804.07998 (2018)

2. Belinkov, Y., Bisk, Y.: Synthetic and natural noise both break neural machine translation. arXiv preprint arXiv:1711.02173 (2017)
3. Cheng, M., Wei, W., Hsieh, C.J.: Evaluating and enhancing the robustness of dialogue systems: A casestudy on a negotiation agent (2019)
4. Ebrahimi, J., Rao, A., Lowd, D., Dou, D.: Hotflip: White-box adversarial examples for text classification. arXiv preprint arXiv:1712.06751 (2017)
5. Gao, J., Lanchantin, J., Soffa, M.L., Qi, Y.: Black-box generation of adversarial text sequences to evade deep learning classifiers. In: 2018 IEEE Security and Privacy Workshops (SPW). pp. 50–56. IEEE (2018)
6. Gao, S., Ren, Z., Zhao, Y., Zhao, D., Yin, D., Yan, R.: Product-aware answer generation in e-commerce question-answering. In: Proceedings of the Twelfth ACM International Conference on Web Search and Data Mining. pp. 429–437. ACM (2019)
7. Goodfellow, I.J., Shlens, J., Szegedy, C.: Explaining and harnessing adversarial examples. arXiv preprint arXiv:1412.6572 (2014)
8. Jia, R., Liang, P.: Adversarial examples for evaluating reading comprehension systems. arXiv preprint arXiv:1707.07328 (2017)
9. Joachims, T.: A probabilistic analysis of the rocchio algorithm with tfidf for text categorization. Tech. rep., Carnegie-mellon univ pittsburgh pa dept of computer science (1996)
10. Li, J., Monroe, W., Jurafsky, D.: Understanding neural networks through representation erasure. arXiv preprint arXiv:1612.08220 (2016)
11. Lowe, R., Pow, N., Serban, I., Pineau, J.: The ubuntu dialogue corpus: A large dataset for research in unstructured multi-turn dialogue systems. arXiv preprint arXiv:1506.08909 (2015)
12. Madry, A., Makelov, A., Schmidt, L., Tsipras, D., Vladu, A.: Towards deep learning models resistant to adversarial attacks. arXiv preprint arXiv:1706.06083 (2017)
13. Mikolov, T., Sutskever, I., Chen, K., Corrado, G.S., Dean, J.: Distributed representations of words and phrases and their compositionality. In: Advances in neural information processing systems. pp. 3111–3119 (2013)
14. Miyato, T., Dai, A.M., Goodfellow, I.: Adversarial training methods for semi-supervised text classification. arXiv preprint arXiv:1605.07725 (2016)
15. Mudrakarta, P.K., Taly, A., Sundararajan, M., Dhamdhere, K.: Did the model understand the question? arXiv preprint arXiv:1805.05492 (2018)
16. Papernot, N., McDaniel, P., Goodfellow, I., Jha, S., Celik, Z.B., Swami, A.: Practical black-box attacks against machine learning. In: Proceedings of the 2017 ACM on Asia conference on computer and communications security. pp. 506–519. ACM (2017)
17. Serban, I.V., Sordoni, A., Bengio, Y., Courville, A., Pineau, J.: Building end-to-end dialogue systems using generative hierarchical neural network models. In: Thirtieth AAAI Conference on Artificial Intelligence (2016)
18. Tao, C., Wu, W., Xu, C., Hu, W., Zhao, D., Yan, R.: Multi-representation fusion network for multi-turn response selection in retrieval-based chatbots. In: Proceedings of the Twelfth ACM International Conference on Web Search and Data Mining. pp. 267–275. ACM (2019)
19. Wu, Y., Wu, W., Xing, C., Zhou, M., Li, Z.: Sequential matching network: A new architecture for multi-turn response selection in retrieval-based chatbots. arXiv preprint arXiv:1612.01627 (2016)
20. Yang, P., Chen, J., Hsieh, C.J., Wang, J.L., Jordan, M.I.: Greedy attack and gumbel attack: Generating adversarial examples for discrete data. arXiv preprint arXiv:1805.12316 (2018)
21. Zhao, Z., Dua, D., Singh, S.: Generating natural adversarial examples. arXiv preprint arXiv:1710.11342 (2017)
22. Zhou, X., Li, L., Dong, D., Liu, Y., Chen, Y., Zhao, W.X., Yu, D., Wu, H.: Multi-turn response selection for chatbots with deep attention matching network. In: Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers). pp. 1118–1127 (2018)