ISSN 1330-3651 (Print), ISSN 1848-6339 (Online)

https://doi.org/10.17559/TV-20200801053744

Original scientific paper

Text Adversarial Examples Generation and Defense Based on Reinforcement Learning

Yue LI*, Pengjian XU, Qing RUAN, Wusheng XU

Abstract: In recent years, the neural networks are widely used in image processing, natural language processing and other fields. But there are new security issues-the adversarial examples. Crafted adversarial examples can make a trouble for the neural network, which leads to the mis-classification. Text classification is one of the basic tasks of the natural language processing. This paper is concerned about the generation and defense of text adversarial examples. The main contributions of this research are as follows: This paper explores a new type of adversarial example and applies reinforcement learning to generate the adversarial examples; a training set composed of adversarial examples is constructed. To build a more robust classifier, a new defense framework is established. In order to eliminate the influence of noise, well-designed predetector and reformer were implemented, which helps the neural networks to resist adversarial examples and reduce coupling.

Keywords: adversarial examples; defense; neural networks; text classification

1 INTRODUCTION

1.1 Background

With the development of neural network, it is applied to many fields of internet security, such as recognizing spams. The neural networks improve the technical level of network security defense, but the emergence of adversarial examples highlights the vulnerability of neural networks. The adversarial example is a carefully crafted input that makes the neural network do something wrong. Fig. 1 is an adversarial example cited from [6]. The panda picture superposing with noise leads the classification model predicts a gibbon picture.

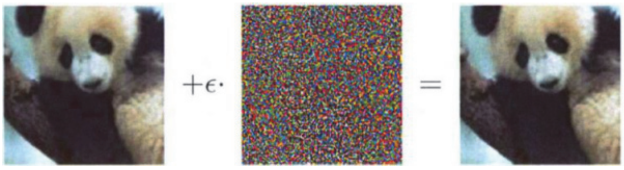

Figure 1 The panda picture superposing with noise leads to a classification mistake

Adversarial examples first appeared in the field of Computer Vision (CV) [1]. After years of development, they have successfully attacked speech recognition systems [2] and reading comprehension systems [3] and they have taken effect in the real world [5]. So the adversarial examples bring a great threat to the neural networks and both the computer vision and the Nature Language Processing (NLP) are facing this security challenge. On the other hand, if the neural networks that are widely used in the security field cannot resist the attack of the adversarial examples, it will lead to more serious security accidents. If the attacker has the ability to generate the adversarial examples that can fool the neural networks, it is more possible for them to break through the defense and more users will suffer from their attacks. Moreover, the great danger of adversarial examples is that they can transfer from one model to another. Therefore, how to deal with adversarial examples and build a more reliable and generalized neural network is a problem urgent to be solved.

Text classification is one of important methods for processing text data. It is widely used in sentiment analysis, public opinion monitoring and other fields. Text classification is also an important method to understand the meaning of text and the emotional tendency of text. For taking up the challenge brought by the text adversarial examples, it is necessary to determine the cause so as to prevent such attacks. Early detection of weak links in neural networks is conducive to establish a more practical and secure system, which lays the foundation for the widespread application of neural network systems.

1.2 Related Work

Neural networks technology is an important part of machine learning. Meanwhile, the adversarial examples are also a risk for machine learning. It is known that dirty data affect the performance [7, 8]. Supervised learning, unsupervised learning and reinforcement learning all get this problem [9]. Generating adversarial examples has low requirements. The black-box attack can still be accomplished even if the attacker knows nothing about the model details [10]. Tramers et al [24] successfully attacked the Logistic Regression when they did not know the model details. But many classic networks are facing this challenge. Even the Alex Net [11] which performs very well in the image classification can also be cheated.

Text classification also has this problem. The text classifier has quite a limited ability to resist the adversarial examples. Adding specific words is an effective approach to generating the adversarial examples [12]. Replacing or removing specific words is also a method [23]. Jia and Liang [4] successfully fool the model in SQuAD reading comprehension task.

To resist the adversarial examples, different kinds of methods are proposed. The ANTIDOTE helps the model to defend the poisoning attack [13]. Clearing the training data is an approach to defend the poisoning attack [14]. Network distillation extracts knowledge to improve the robustness of neural networks, which is tested on the MNIST [15] and CIFAR-10 [16] dataset and gets success. If the classifier is robust enough, it can make fewer mistakes when processing the adversarial examples. So it is reasonable for researchers to build a more robust classifier to defend the adversarial examples. Bradshaw et al. [17] change the structure of the CNN and improve its performance. Many tools, like auto-encoders [18], detection sub- networks, and dropout units [19], can detect the adversarial examples. Based on the feature of adversarial examples, the specific CNN models that pay more attention to the relative class have a more competitive performance [20].

In summary, the research on the attack and defense has great development as the new defense methods are found [21].

1.3 Contributions

This paper aims at the text adversarial examples generation and defense. The contributions are included:

(1) English is an alphabetic language where the space is a good approximation of a word delimiter. Meanwhile, the hyphen is used to join words to indicate that they have a combined meaning. So replacing specific spaces with hyphens significantly changes the text semantics. This paper discovers a new type of text adversarial example based on this feature.

(2) The main difficulty of generating is how to determine the position of the replacement space. This paper proposes to use reinforcement learning to solve the generation problem and build an adversarial examples dataset. The experimental results indicate that this adversarial example can successfully fool the neural network.

(3) This paper builds a new defense framework to resist adversarial example attack. Its structure has a bastion layer that acts as a gateway between the input and the classification model. The bastion layer, made up of detector and a reformer, can remove the noise information in the adversarial example and protect the classification model. And the defense framework is a way to reduce coupling. In other words, the detector, the re-constructor and the classification model can be optimized respectively.

2 PRELIMINARY WORK

This section introduces some preliminary work. The text sentiment classification is introduced as the essential research for adversarial examples problem.

2.1 Dataset

The IMDb dataset [22] is a data set for binary sentiment classification, which contains 50,000 movie reviews. There are two types of reviews, one is the positive reviews and the other is the negative reviews. The distribution of the reviews is balanced. There are 12,500 positive reviews and 12,500 negative reviews for training. And test set is similar to the training set, which also contains 12,500 positive reviews and 12,500 negative reviews. In this paper, the basic task is to classify the IMDb data set correctly.

2.2 Sentiment Classification Description

The ground-truth classifier is donated by:
f: $X \rightarrow Y$. f maps every input x to a label y where $x \in X$ and $y_i = f(x_i)$. In this paper, the input is the movie review and the output is the types of emotions. Define the dataset: $D = \{x_i, y_i\}_{i=1}^{n}$.

The emotion classification is a binary classification which includes the positive class and the negative class. f^+ represents the positive class and f^-represents the negative class. The x is donated by x^+ when $f(x) = x^+$. That means the review is positive. The x is donated by x^- when $f(x) = f^-$, which indicates the review is a negative one. In summation, when doing sentiment classification, the goal is to find the suitable classifier h that satisfies:

$$\begin{cases} h\left(x^+\right) = f^+ \\ h\left(x^-\right) = f^- \end{cases} \tag{1}$$

Eq. (1) means both positive and negative reviews can be correctly classified by the classifier. How to build this classifier is introduced in the section 2.3.

2.3 Sentiment Classification Details

Raw data records the nature language that cannot be directly processed by computer. The computer only processes the numbers and vectors. Text representation method transfers the raw data to word embedding and plays an important role in the classifier performance. Therefore, the word2vec is selected. The word2vec is a popular method used in neural language process, which has better performance on producing the vectors to represent the words comparing with other methods.

The word2vec is the expression of the corresponding word in a multidimensional space and measures the semantic distance between different words, so it can express the meaning of the text very well. The word embedding transfers the text to vectors whose magnitude represents the meaning of the text. Comparing with the bag-of-words model, word2vec expresses semantics more accurately and overcomes the semantic gap problem. Meanwhile, word2vec does not miss the combination of neighboring word for the reason that the training samples utilize the combination of neighboring word. The word embedding constructed by word2vec is a dense vector, but not a sparse vector, which means it contains more information in the same dimension.

Word2vec has two computing paradigms, one of them is the continuous bag-of-words model (CBOW), and the other is the skip-gram model. This paper uses the word-skip model to train word embedding. The training data for word2vec is all reviews in the IMDb and the dimension of word embedding is set to 400. So the hidden layer in the word2vec training model has 400 neurons. The cosine similarity between the predicted headword and the five background words is used as the loss function.

Movie review is a kind of variable-length text. The length of every review is different. The word2vec only provides the word embedding representing the feature of text. To ensure that the input is a fixed-length vector, the feature of every review is calculated by taking the average of concatenating the word embedding. Then, the input is processed by the classification model, including Naive Bayes (NB), Support Vector Machine (SVM), Multilayer Perceptron (MLP), Convolutional Neural Network (CNN), Recurrent Neural Network (RNN), Long Short-Term Memory Network (LSTM), and Gating Circulation Unit

(GRU). NB is a generative approach. SVM and neural network are the discriminative approach. MLP, CNN, RNN, LSTM and GRU are typical neural networks. Experiments show that the classification models can effectively classify the positive reviews and negative reviews. The experimental results are presented in Section 5.

3 ADVERSARIAL EXAMPLES GENERATION

Based on the preliminary work, a new type of text adversarial example is designed and presented in the following. Moreover, reinforcement learning will be applied to solve the problem of text adversarial example generation.

3.1 Attack Description

Adversarial example a is the movie review x after modifying. The crafted x^+ is donated by a^+ and the crafted x^- is donated by a^-. a^+ means its original text x is a positive review(x^+) a^- means its original text x is a negative review(x^-). Adversarial example a can lead to a classification mistake, meaning the a satisfy:

$$\begin{cases} h\left(a^+\right) = f^+ \\ h\left(a^-\right) = f^- \end{cases} \tag{2}$$

Eq. (2) describes the attack target of the adversarial examples, which is to fool the classifier.

As a result, the adversarial example a^+ is classified as a negative review while its original text is a positive review (x^+). And the a^- is classified as a positive review while it is generated by modifying a negative review. So, adversarial examples bring great challenge to the classification model that is applied in many fields.

English is an alphabetic language where every word is split by the spaces. But this feature is the weak point that gives the attacker a chance. The hyphens are a typical symbol used in English, which joins several words together to indicate that they are in a combined meaning. And the hyphenation is an arbitrary rule for English. So we get the inspiration that replacing the spaces with hyphens is a method to produce the adversarial examples. There is an example shown in Tab. 1.

Table 1 Raw data and adversarial example

Input	Content
Original text	Busy Phillips put in one hell of a performance, both comedic and dramatic. Erika Christensen was good but Busystoletheshow.Itwas a nice touch after The Smokers, a movie starring Busy, which wasn't all that great. If Busy doesn't get a nomination of any kind for this film it would be a disaster. Forget Mona Lisa Smile, see HomeRoom.
Adversarial example	Busy Phillips-put in one-hell of a performance, both-comedic-and-dramatic. Erika Christensen was good-but Busy-stole the show. It-was a nice touc hafter The Smokers, a moviestarring-Busy, whichwasn't all-that great. If-Busy doesn't get-a nomination of-any kind for this- film-it would be a disaster. Forget Mona Lisa Smile, see-Home Room.

The original text is a positive review selected from the IMDb dataset and the classification model can do the right classification. But the adversarial example fools the classification model after replacing specific spaces in the raw data.

The original text is a positive review selected from the IMDb dataset and the classification model can do the right classification. But the adversarial example fools the classification model after replacing specific spaces in the raw data.

3.2 Adversarial Example Generation Based on Reinforcement Learning

The adversarial example mentioned above is not a great obstacle to reading. And it would be easy for users to ignore if the user did not check carefully. Therefore, this kind of adversarial example has good stealthiness and is hard to be detected, which is similar to the adversarial example in the image field.

So the most important problem in adversarial example generation is to identify which spaces should be replaced. A reinforcement learning method is designed and implemented to solve this problem.

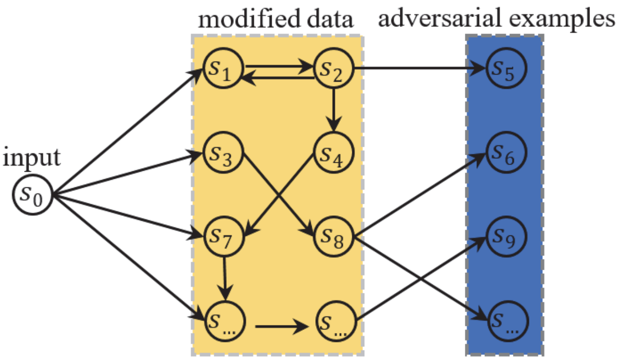

Figure 2 Adversarial example generation based on reinforcement

Fig. 2 shows the process of generating the adversarial examples. This paper assumes a virtual agent. The state of agent changes after modifying the text. The raw data is the state s_0, in which no spaces are replaced. The agent goes to a new state after replacing specific spaces. These states are in the yellow and blue region in Fig. 2. For a classification model, the raw data may generate multiple adversarial examples after replacing different spaces with hyphens, which leads to the classification mistakes. Hence an original text may correspond to multiple states, that is, any state in the blue region is an adversarial example. The main objective is to find an appropriate state in the blue region and output as an adversarial example.

3.3 Reward Derivation

It is necessary to define a loss function to evaluate the difference of the raw data and the adversarial examples. The loss function is applied to guide the agent action. This loss function is the reward for reinforcement learning. The following is a detailed introduction to the reward derivation. Generating adversarial example requires querying the output of the classification model. It takes a lot of time to query the output when the model is complex. Therefore, this paper chooses a shallow MLP as the basic sentiment classification model to generate the adversarial example. Fig. 3 shows the structure of an MLP. The MLP is a simple neural network that is inspired by the action mechanism of biological nerves. The information processed by the neurons in each layer is passed to the next layer. The neurons are fully connected to every neuron in the next layer.

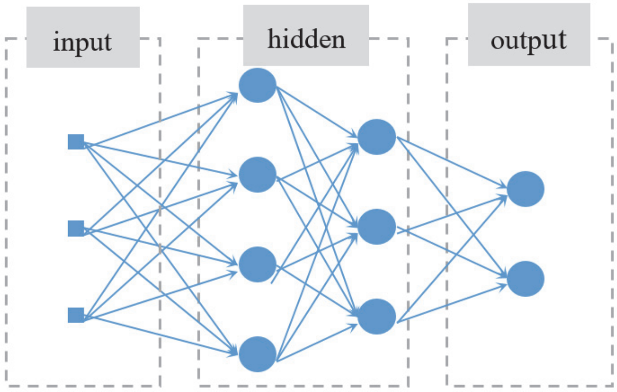

Figure 3 The MLP structure

A shallow MLP has only a few of the parameters. So the computational amount is small and it takes less time to calculate the output. In summary, the shallow MLP meets the demands of quickly calculating the model outputs. The output layer is the softmax layer that outputs a separate probability for the two types of sentiment. The formula for softmax is:

$$S_i = \frac{e^i}{\sum e^j} \tag{3}$$

In this paper, the sentiment classification is a binary classification problem: positive or negative. So the output of the softmax layer is two probabilities, which is denoted by:

$$h(x) = \left[p^+ p^- \right] \tag{4}$$

p^+, p^- are the probability of the output layer. p^+ is the probability of being recognized as a positive review and p^- is the probability of being recognized as a negative review. According to Eq. (3), the following equation is obtained:

$$p^+(x) + p^-(x) = 1 \tag{5}$$

The label of the positive class is (1, 0) and the label of the negative class is [0, 1]. That is to say:

$$\begin{cases} f^+ = (10) \\ f^- = (01) \end{cases} \tag{6}$$

The loss function is applied to measure the distance between the original text and the adversarial text:

$$L(x) = \mathrm{e}^{-f(x)h(x)} \tag{7}$$

For a positive review x^+, the ideal output is negative when to generate the adversarial example, which means the crafted review fools the classification model. So we compute the loss function based on Eq. (4), Eq. (6) and Eq. (7) and obtain:

$$\begin{aligned} L\left(x^+\right) &= \mathrm{e}^{-f^-\left(x^+\right)h\left(x^+\right)} = \mathrm{e}^{-f^- \cdot h\left(x^+\right)} = \\ &= \mathrm{e}^{-f^-(0\ 1)\cdot\left[p^+\left(x^+\right)p^-\left(x^+\right)\right]} = \mathrm{e}^{-p^-\left(x^+\right)} \end{aligned} \tag{8}$$

Eq. (8) shows that $L(x^+)$ decreases as the $p^-(x^+)$ becomes larger, which means that the positive review is more likely to be treated as a negative review. So it is reasonable to simplify the loss function:

$$L\left(x^+\right) = p^-\left(x^+\right) \tag{9}$$

For a negative review, the loss function is simplified in the same way:

$$L\left(x^-\right) = p^+\left(x^-\right) \tag{10}$$

The loss function becomes very simple and serves as the reward in the reinforcement learning, which guides the agent in what action to take. For a positive review, the loss guides the agent to transfer a positive review to a negative review. But it cannot decide what concrete action to take. More details about how to produce the adversarial examples are shown in algorithm 1. When the reward becomes larger, the agent replaces spaces with hyphens. When the reward becomes smaller, the agent replaces the hyphens with spaces. The negative reviews are transformed into adversarial examples after the same process.

Algorithm 1 The adversarial examples generation method

```
IInput: a quaternary $E = (S, A, P, R)$
Process:
  $L(s) = 1, L(s^j) = 1$
  $T$ is the number of spaces in original state $s_0$
  while $T > 0$ and $L(s) > \varepsilon$ :
    if $L(s^j)\ L(s) <= 0$ :
      random chose spaces then replace them with hyphens
    if $L(s^j)\ L(s) > 0$ :
      random chose hyphens then replace them with spaces
Output: current states'
```

The algorithm 1 is an approach to generate the adversarial examples. The MLP network serves as the environment for the agent. The agent gets a reward from the MLP after modifying the input text. Then the agent takes some action guided by the reward. When the reword becomes larger, the agent automatically finds the right place to replace the blanks with the hyphens. When the reword becomes smaller, the agent randomly replaces the hyphens with the blanks. When the agent satisfies the demands of the adversarial examples, it stops trying and the next agent will do the same thing. After this process, an original text becomes an adversarial example. Comparing to other reinforcement learning methods, the most difference is that the agent's state is the focus in this paper. The finial output is a state of the agent.

3.4 Build Adversarial Set

Producing the adversarial examples is a time-consuming task. Therefore, we select a subset of the IMDb to produce the adversarial examples, which is called the part test set. The reviews in the part test set are selected from the test part of the IMDb. It includes 800 positive reviews and 800 negative reviews. The adversarial set is generated based on the part test set. The part test set uses systematic sampling to get a fair representation of the IMDb test set. The IMDb data set is divided into 100 groups and then a review is selected from every group. So every subfile has 100 reviews. And we select 8 subfiles to build the part test set.

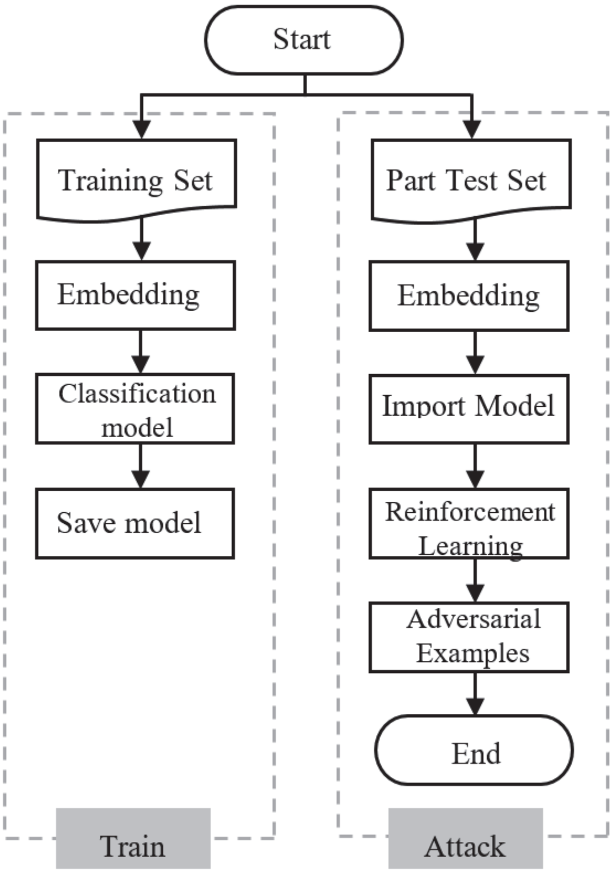

Figure 4 The adversarial set building process.

Fig. 4 shows the whole process of generating adversarial examples. The generation process can be divided into two stages. The first stage is the training stage. The main task is to build a text classification model. In the training stage, the classification model learns from the data in the training set. Then the model is saved and will be used in the reinforcement learning. The classification model will serve as the environment for reinforcement learning. In the attacking stage, the algorithm 1 will be applied to generate adversarial examples. The classification model parameters remain unchanged, and the agent explores how to generate adversarial examples under the guidance of $L(x)$. At this stage, the agent continuously tries to replace the spaces in different positions and finally determines an agent state that can fool the classification model. Finally, all reviews in the part text set are processed and they constitute the adversarial set. The adversarial set is the important data set to test the adversarial examples attack results.

4 ADVERSARIAL EXAMPLES DEFENCE

4.1 Defend Description

Under attack of the adversarial examples, the model cannot do classification correctly. Their performance decreases a lot. The adversarial example is a great security threat, which has a great impact on the entire system. Therefore, it is necessary to build a more robust model to resist the adversarial examples. If a new classification model H can be constructed, it satisfies Eq. (11):

$$\begin{cases} H\left(x^{+}\right)=f^{+} \\ H\left(x^{-}\right)=f^{-} \end{cases} \tag{11}$$

That is the model that correctly classifies the positive reviews and negative reviews. At the same time, the new model satisfies Eq. (12):

$$\begin{cases} H\left(a^{+}\right)=f^{+} \\ H\left(a^{-}\right)=f^{-} \end{cases} \tag{12}$$

Eq. (12) means the model H succeeds to resist the adversarial examples.

Eq. (11) and Eq. (12) are two important constraints for a robust model. It measures whether the model resists the adversarial examples. In other words, the ensemble modes classify the original text correctly and have excellent performance on the adversarial examples.

4.2 Noise Disturbance

Take the discriminative classification model to solve the binary classification problem for an example. The training process is to determine a hyperplane by learning the training data, where different sides of the hyperplane are different types of data. If an input x was transferred into an adversarial example, it would be in the other side of the hyperplane. So the prediction of the classification model is also changed. The adversarial example is the original text combined with a small perturbation, which has a bad effect on the performance of the classification model. Such small disturbances are called noise. As it is shown in Fig. 5, the original text x combining with the noise becomes a different input x'. The noise causes x to pass through the classification hyperplane and the x' to the other side of the hyperplane. So, the noise changes the distribution of the original text and fools the classification model. Even if the classification hyperplane is optimal and can correctly classify the original text, with the influence of noise, some texts still pass through the hyperplane and enter another category distribution area. At the same time, the classification hyperplanes built by most models are not optimal. Therefore, the crafted input with a little change may pass through the classification hyperplanes.

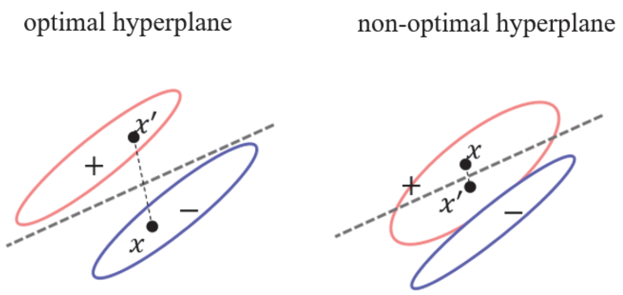

Figure 5 Noise has a bad effect on the model performance

4.3 Multilayer Structure Defense Framework

According to the generation process, adversarial example is a combination of the original text and noise. Well-designed noise is an important reason for incorrect classification. The noise makes the adversarial examples deviate from the original text. They are so different that they pass through the classification hyperplane. If the noise is removed from the adversarial examples, the distribution of the original text does not change and the adversarial examples do not pass through the hyperplane. Therefore, the noise has no influence on the performance of the classification model, which helps the classification model to resist the attack of the adversarial example. So, there is a solution inspired by such idea.

Fig. 6 shows the structure for defense framework. This framework adds a bastion layer before the classification model, which is consisted of a detector and a reformer. The input data is firstly processed by the detector to recognize whether it is an adversarial example. If the detector finds an adversarial example, it is passed to the classification model after being processed by the reformer. But the input, which is not recognized by the detector as adversarial example, is directly passed to the classification model.

The detector has to distinguish the adversarial examples from the input and pass them to the reformer. So, the detector checks all the input text. The detector makes demands of simple structure, low time complexity and fast calculation speed to reduce the time consuming. The required time to generate the adversarial example is long. And the quantity of the data used to train the detector is relatively small. Therefore, the detector needs to learn quickly when the training data is small so as to effectively detect the adversarial examples.

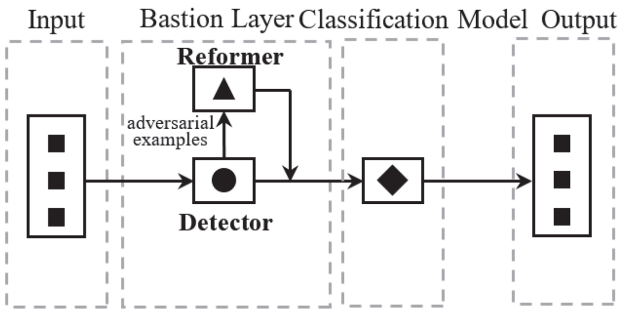

Figure 6 Defense framework structure

The reformer is to remove the noise from the adversarial example and transfer the adversarial example to the original text. Hence the adversarial example returns to the correct side of the hyperplane. The reformer only processes the input recognized as the adversarial example by the detector. In other words, the reformer only processes a part of the input. The reformer replaces the hyphens with spaces to remove the noise and then passes the noise-removed input to the classification model. Therefore the data processed by the classification model is very similar to the original text. Tab. 2 shows an example of the reformer output. The reformer replaces the hyphens in the adversarial example with the space.

Table 2 Data processed by the reformer

Category	Content
Adversarial example	Busy Phillips-put in one-hell of a performance, both-comedic-and-dramatic.Erika Christensen was good-but Busy- stole the show. It-was a nice touch after The Smokers, a movies tarring-Busy, which wasn't all-that great. If-Busy doesn't get-a nomination of-any kind for this-film-it would be a disaster. Forget Mona Lisa Smile, see-HomeRoom.
Processed by the reformer	Busy Phillips put in one hell of a performance, both comedic and dramatic. Erika Christensen was good but Busy stole the show. It was a nice touch after The Smokers, a movie starringBusy, which wasn't all that great. If Busy doesn't get a nomination of any kind for this film it would be a disaster. Forget Mona Lisa Smile, see HomeRoom.

It is necessary to build a detector, which ensures the Bastion layer has little influence on the input. Because the reformer processes all the input data with no detector, it affects the input too much. The original text is modified even it is not an adversarial example and has no noise. So adding a detector is to reduce the modification of the input. The reformer only takes effect when the input is determined as the adversarial example. In contrary, the input determined as the original text is directly passed to the classification model. The reformer does not affect the original text.

4.4 Build Detector and Reformer

The bastion layer consists of the detector and the reformer, which is the first line of defense against adversarial examples. They are similar with the firewall, which separates the security function from the sentiment classification function in order to decouple different parts efficiently. The main difficulty of the detector is the binary classification including adversarial examples and original text. The train data in the experiment contains 2500 adversarial examples and 2500 unmodified reviews. This data set is applied for training the detector, which is a simple MLP with the same structure as the MLP used for sentiment classification. The Reformer filters the input text transferred from the detector. That is, the space replaced by the hyphen is restored which makes the input text contain no noise. Then the input text processed by the reformer is transferred to the classification model for sentiment classification. The reformer replaces underscores, hyphens, and other symbols in the text with spaces so as to ensure the input still separates words by spaces. In this way, the reformer protects the classification model from noise in the input.

In order to determine whether the bastion layer can effectively help the classification model to resist the adversarial examples, this defense framework is tested on the part test set and the adversarial test set. The part test set is to evaluate whether the defense framework is able to correctly classify the original text, while the adversarial set is to evaluate whether the defense framework is able to resist the adversarial example. Under ideal condition that the bastion layer helps the classification model to resist the adversarial examples, the classification model has a good performance on both the part test set and the adversarial set. The defense framework can correctly classify the adversarial examples without being affected by the noise they carry.

5 EXPERIMENTS

5.1 Accuracy Evaluation Indexes

The sentiment classification is a typical binary classification problem. According to the data label and the prediction made by the classification model, the confusion matrix in Tab. 3 can be obtained.

Table 3 Confusion matrix

		True label	
		Positive	Negative
Prediction	Positive	*TP*	*FP*
	Negative	*FN*	*TN*

TP is the case in which the model correctly classifies positive reviews as positive reviews. *FP* is the case in which the model classifies negative reviews as positive reviews. *FN* are the cases in which the model classifies positive reviews as negative reviews. And *TN* are the cases in which the model correctly classifies negative reviews as negative reviews. The accuracy can be defined as the percentage of correctly classified instances:

$$Auc = \frac{TP + TN}{TP + TN + FP + FN} \tag{13}$$

The accuracy measures whether the classification model can correctly classify the input. The model performance is better when the accuracy is higher. In the IMDb dataset, the amount of positive reviews is the same as negative reviews. So the distribution is balanced. The accuracy can well reflect the performance of the model. If the accuracy rate is high, the model has a good performance on the classification task. Otherwise, it misclassifies positive reviews as negative reviews or misclassifies the negative reviews.

5.2 Basic Sentiment Classification Result

To measure the quality of the test set, different methods are used to conduct the sentiment classification. As shown in Tab. 4, both machine learning models and neural networks finish the classification task. The NB has the worst performance with an accuracy rate of 67.46%. On the other hand, the accuracy of other models is all higher than 82% except NB. Different models have different performance on the sentiment classification. The performance of the neural network is better than the machine learning method on the whole. The LSTM performances are the best, whose accuracy is 85.13%.

In addition, the accuracy of the test set is close to the part test set. For different methods, the performance on the test set is similar to the performance on the part test set. The difference is never beyond 3%, which indicates that the part test set is a typical representation of the test set. In the next subsection, the part test set will represent the test set to measure the performance of different models for the reason that the adversarial set has limited reviews.

Table 4 Performance of different methods on the test set and the part test set

Sub-field	Model	Test set	Part test set
Machine learning	NB	67.46%	67.69%
	SVM	83.66%	85.19%
Deep learning	MLP	82.56%	84.00%
	CNN	83.93%	85.23%
	RNN	84.23%	85.81%
	LSTM	85.13%	86.46%
	GRU	85.06%	86.39%

5.3 Attack Result

To evaluate the influence of the adversarial examples on the classification models, different machine learning methods and neural networks are applied to classify the data in the adversarial set. These methods include NB, SVM, MLP, CNN, RNN, LSTM, and GRU.

Table 5 The attack result on different models

Sub-field	Model	Part test set	Adversarial test set
Machine learning	NB	67.69%	48.31%
	SVM	85.19%	52.06%
Deep learning	MLP	84.00%	51.50%
	CNN	85.23%	52.23%
	RNN	85.81%	53.45%
	LSTM	86.46%	53.32%
	GRU	86.39%	53.78%

Tab. 5 shows the attack result. The experiment result shows that the adversarial examples have a very bad influence on the performance of all methods tested in this paper. The classification accuracy decreases significantly when confronting with adversarial examples. Under the attack of the adversarial examples, the classification accuracy is close to 50%. This means that the parameters are almost randomly initialized. All methods are vulnerable when attacked by adversarial examples. In addition, the adversarial example is generated based on MLP, but it has a significant effect on other neural networks and machine learning models.

In general, this new type of adversarial example has great influence on the model classification performance. At the same time, model performance matters whether the models are worth to be applied in practice. But the adversarial examples bring great risks to machine learning and neural network.

5.4 Defense Result

Tab. 6 shows the comparison of the two schemes. The basic classifiers include NB, SVM, MLP, CNN, RNN, LSTM, and GRU. The defense framework is these classifiers with a bastion layer, which consist of the detector and the reformer. It is shown that the defense framework and the base classifier have a similar performance on the part test set. So the part test set is a good representation of the test set.

The detectors and reformer added to the basic classifier in defense framework do not have a serious negative impact on the classification accuracy of the classification models. And the classification accuracy of some classification models decreases a little.

Table 6 The performance of the basic classifiers and the defense framework on the part test set

Sub-field	Model	Basic classifier	Defense framework
Machine learning	NB	67.69%	67.96%
	SVM	85.19%	85.28%
Deep learning	MLP	84.00%	85.74%
	CNN	85.23%	85.44%
	RNN	85.83%	85.41%
	LSTM	86.46%	86.07%
	GRU	86.39%	86.32%

To verify whether the model can effectively resist the security threats brought by the adversarial examples there is a performance comparison in Tab. 7.

Table 7 The performance of the basic classifiers and the defense frame- work on the adversarial set

Sub-field	Model	Basic classifier	Defense framework
Machine learning	NB	48.31%	57.66%
	SVM	52.06%	75.53%
Deep learning	MLP	51.50%	75.52%
	CNN	52.23%	75.54%
	RNN	53.45%	75.65%
	LSTM	53.32%	75.59%
	GRU	53.78%	75.91%

Tab. 7 shows that the performance of the defense framework on the adversarial set is significantly better than the basic classifier, which means the ability of each classification model to resist adversarial examples is significantly enhanced. The classification accuracy of machine learning algorithms on adversarial example sets has been improved by more than 9%. The classification accuracy of the neural network on the adversarial examples has been improved by more than 20%.

6 CONCLUSION AND FUTURE WORK

English word segmentation is based on the spaces. But this character brings great risk, which can be used to generate a new type of adversarial examples based on the reinforcement learning method proposed in this paper. And this new type of adversarial examples successfully attacks the text classification models. It is not necessary to understand the model structure to generate this type of adversarial examples. So the attack is a black box attack. Adversarial examples generated based on one neural network can also effectively attack not only neural networks but also traditional machine learning algorithms. The defense framework proposed in this paper can effectively resist the adversarial examples attack. It helps the classification models defense adversarial examples.

There are several problems that need to be investigated in the future. This paper only generates English adversarial examples. But English is an alphabetic language that splits the words by spaces. However, Chinese is a kind of the logogram which is totally different compared with English. Whether or not the neural networks applied to Chinese language processing have the same weak points is also a question. The adversarial examples are designed for the classification problem. Further research is needed to confirm the influence of the adversarial examples on machine translation, man-machine dialogue and other fields. The defense framework helps the classifier to resist the adversarial examples. But it is still unable to completely resist the attack of adversarial examples. It is necessary to develop a better defense method, which can resist adversarial examples and correctly classify the original text. Moreover, we have the faithful belief that better defense method will be proposed in the future.

7 REFERENCES

[1] Brundage, M., Avin, S., Clark, J., et al. (2018). The malicious use of artificial intelligence: Forecasting, prevention, and mitigation. http://arxiv.org/abs/1802.07228

[2] Carlini, N., Mishra, P., et al. (2016). Hidden voice commands, in: 25th USENIX Security Symposium (USENIX Security 16), Austin, TX. 513-530.

[3] Jia, R. & Liang, P. (2017). Adversarial examples for evaluating reading comprehension systems. *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing, Copenhagen, Denmark*, 2021-2031. https://www.aclweb.org/anthology/D17-1215. https://doi.org/10.18653/v1/D17-1215

[4] Jia, R. & Liang, P. (2017). Adversarial examples for evaluating reading comprehension systems. *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing, EMNLP, Copenhagen, Denmark*, 2021-2031. *https://doi.org/10.18653/v1/D17-1215*

[5] Sharif, M., Bhagavatula, S., Bauer, L., & Reiter, M. K. (2016). Accessorize to a crime: Real and stealthy attacks on state-of-the-art face recognition. Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security, ACM, New York, NY, USA, 1528-1540. https://doi.org/10.1145/2976749.2978392

[6] Goodfellow, I., Shlens, J., & Szegedy, C. (2015). Explaining and harnessing adversarial examples. *International Conference on Learning Representations*. http://arxiv.org/abs/1412.6572.

[7] Biggio, B., Nelson, B., & Laskov, P. (2012). Poisoning attacks against support vector machines, Edinburgh, United kingdom. 1807-1814.

[8] Biggio, B., Pillai, I., Rota Bulo, S., Ariu, D., Pelillo, M., & Roli, F. (2013). Is data clustering in adversarial settings secure? Berlin, Germany. 87-97. https://doi.org/10.1145/2517312.2517321

[9] Pattanaik, A., Tang, Z., Liu, S., Bommannan, G., & Chowdhary, G. (2018). Robust deep reinforcement learning with adversarial attacks. *Proceedings of the 17th International Conference on Autonomous Agents and Multi-Agent Systems, Richland, SC*. 2040-2042.

[10] Papernot, N., McDaniel, P., Wu, X., Jha, S., & Swami, A. (2016). Distillation as a Defense to Adversarial Perturbations against Deep Neural Networks. *Proceedings of IEEE Symposium on Security and Privacy (SP),* 582-597. https://doi.org/10.1109/SP.2016.41

[11] Krizhevsky, A., Sutskever, I., & Hinton, E. G. (2017). Imagenet classification with deep convolutional neural networks. *Communications of the ACM 60*, 84-90. https://doi.org/10.1145/3065386

[12] Li, B. L., Hongcheng, L., Miaoqiang, S., Pan, B., Xirong, L., & Wenchang, S. (2018). Deep text classification can be fooled, Stockholm, Sweden. 4208-4215. https://doi.org/10.24963/ijcai.2018/585

[13] Rubinstein, B. I., Nelson, B., et al. (2009). Antidote: Understanding and defending against poisoning of anomaly

detectors. *Chicago, IL, United states*. 1-14. https://doi.org/10.1145/1644893.1644895

[14] Nelson, B., Barreno, M., et al. (2009). Misleading learners: Co-opting your spam filter. 17-51. https://doi.org/10.1007/978-0-387-88735-7_2

[15] Lecun, Y., Bottou, L., Bengio, Y.,& Haffner, P (1998). Gradient-based learning applied to document recognition. *Proceedings of the IEEE 86*, 2278-2324. https://doi.org/10.1109/5.726791

[16] Krizhevsky, A. (2012). Learning multiple layers of features from tiny images.University of Toronto.

[17] Bradshaw, J., Matthews, A. G. d. G., & Ghahramani, Z. (2017). Adversarial Examples, Uncertainty, and Transfer Testing Robustness in Gaussian Process Hybrid Deep Networks. *arXiv e-prints* , arXiv:1707.02476arXiv:1707.02476.

[18] Huster, T. P., Chiang, C. Y. J., Chadha, R., & Swami, A. (2019). Towards the development of robust deep neural networks in adversarial settings. *Los Angeles, CA, United states*. 419-424. https://doi.org/10.1109/MILCOM.2018.8599814

[19] Feinman, R., Curtin, R. R., Shintre, S., & Gardner, A. B. (2017). Detecting Adversarial Samples from Artifacts. *arXiv e-prints* arXiv:1703.00410.

[20] Abbasi, M. & Gagne, C. (2017). Robustness to adversarial examples through an ensemble of specialists. *Proceedings of the 5th International Conference on Learning Representations Work- shop, ICLR, Palais des Congre`s Neptune, Toulon, France*. 1-9. https://doi.org/10.1145/1014052.1014069

[21] Barreno, M., Nelson, B., Joseph, A. D., & Tygar, J. (2010). The security of machine learning. *Machine Learning 81*, 121-148. https:// doi.org/10.1007/s10994-010-5188-5

[22] Maas, A. L., Daly, C., Pham, P. T., Huang, D., Andrew, J. N., & Potts, C. (2011). Learning word vectors for sentiment analysis. *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies - Volume 1, Association for Computational Linguistics, Stroudsburg, PA, USA*. 142-150. http://dl.acm.org/citation.cfm?id=2002472.2002491

[23] Samanta, S. & Mehta, S. (2017). Towards crafting text adversarial samples. *CoRR* abs/1707.02812. http://arxiv.org/abs/1707.02812,

[24] Tramer, F., Zhang, F., Juerls, A., Reiter, M. K., & Ristenpart, T. (2016). Stealing Machine Learning Models via Prediction APIs. *Proceedings of the 25th Usenix Security Symposium, Usenix. USA*. 601-618.

Contact information:

Yue LI
(Corresponding author)
College of Computer Science and Technology, Donghua University,
Shanghai 201600, China
Email: frankyueli@dhu.edu.cn

Pengjian XU
College of Computer Science and Technology, Donghua University,
Shanghai 201600, China

Qing RUAN
College of Computer Science and Technology, Donghua University,
Shanghai 201600, China

Wusheng XU
College of Computer Science and Technology, Donghua University,
Shanghai 20160f0, China