Journal on Artificial Intelligence
DOI:10.32604/jai.2021.016305
Article

Tech Science Press

A Generation Method of Letter-Level Adversarial Samples

Huixuan Xu[1], Chunlai Du[1], Yanhui Guo[2,*], Zhijian Cui[1] and Haibo Bai[1]

[1]School of Information Science and Technology, North China University of Technology, Beijing, 100144, China
[2]Department of Computer Science, University of Illinois Springfield, Springfield, USA
[*]Corresponding Author: Yanhui Guo. Email: yguo56@uis.edu
Received: 29 December 2020; Accepted: 22 March 2021

Abstract: In recent years, with the rapid development of natural language processing, the security issues related to it have attracted more and more attention. Character perturbation is a common security problem. It can try to completely modify the input classification judgment of the target program without people's attention by adding, deleting, or replacing several characters, which can reduce the effectiveness of the classifier. Although the current research has provided various methods of perturbation attacks on characters, the success rate of some methods is still not ideal. This paper mainly studies the sample generation of optimal perturbation characters and proposes a character-level text adversarial sample generation method. The goal is to use this method to achieve the best effect on character perturbation. After sentiment classification experiments, this model has a higher perturbation success rate on the IMDB dataset, which proves the effectiveness and rationality of this method for text perturbation and provides a reference for future research work.

Keywords: Perturbation attack; sentiment analysis; adversarial examples

1 Introduction

With the rapid development of Internet technology and more and more abundant information content, which greatly enriches people's study, work, and life, the Internet has become an important information acquisition channel in people's work and life. Then, people are faced with the threat of cyberspace security, which involving network security, system security and content security [1–2]. The nature of the semantic meaning carried by data information on the network has always been the object of information content security research. In recent years, with the rapid development of deep learning, especially the large-scale deployment of various neural network models in practical systems such as face recognition, machine translation and text content recognition, the security of these application systems based on deep-learning theories getting more and more attention.

The attacker builds adversarial samples for deep learning and uses them as a data set for the artificial intelligence system to learn and finally realize a deep learning framework trained on the wrong data set. This deliberately constructed "wrong sample" is called an adversarial sample, and this concept was first proposed by Szegedy et al. [3]. In order to deceive classification model of target system, the process which applies a slight disturbance to the original training data set is called adversarial attack. Adversarial attacks can expose the security vulnerabilities of machine learning models and provide support for researchers to subsequently improve the robustness of these model [4]. Although adversarial samples were originally used in the image field, as the research progressed, researchers found that adversarial samples can be expanded to other research field concerning with artificial intelligence. In theory, high-dimensional data of almost all known machine learning models is faced with the risk of being tainted by adversarial examples. The difference after adding disturbance to the training sample is difficult for humans to "perceive", but the deep learning model can perceive the difference caused by this disturbance, which eventually results in the wrong classification

of the data to be recognized [5]. Therefore, not only does it theoretically have the possibility of using "junk samples" to pass the identification classification system, but it even further causes serious security consequences for platforms that use deep learning for identification.

To counter the content detection system based on deep learning, the attacker disturbs the text of bad information, that is, by adding characters and numbers, the attacker slightly changes the bad keywords, and finally forms the recognition of the content detection system without affecting the readers' reading comprehension. It even interferes with the judgment of sentiment classification system used for monitoring data content, so that it can recognize positive sentiment as negative sentiment, as shown in the following Fig. 1 [6].

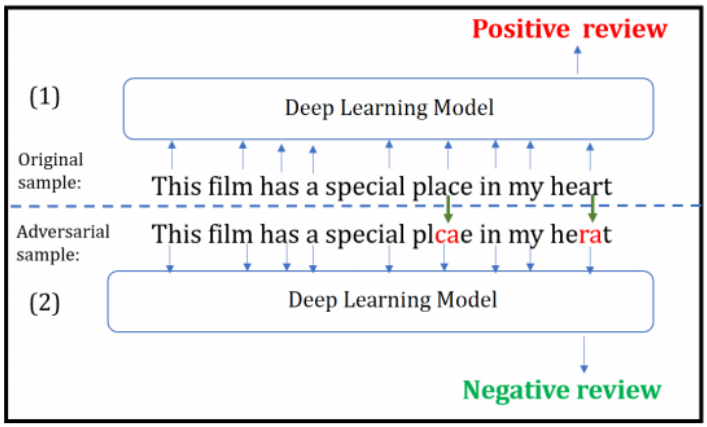

Figure 1: Text adversarial samples [6]

The "statement (1)" is the original sample, while the "statement (2)" is the adversarial sample obtained after several character transformations. It can be seen from Fig. 1 that the positions of two groups of letters {a, c} and {a, r} in 'place' and 'heart' are exchanged through slight changes to some letters. Although the meaning of the sentence can be read and understood from the perspective of the readers themselves, the reader does not even find the spelling differences between "place" and "place", "heart" and "herat", so it does not affect the reading and understanding. However, for the system based on intelligent content analysis, this change makes the normally positive emotion become negative emotion. Therefore, this misclassification of emotions may indirectly lead to other problems.

The contributions made in this paper are as follows: (1) we propose a character-level text adversarial sample generation method. (2) based on the IMDB data set, the proposed method is tested for the recognition of changes in emotional word attributes, and the experimental results prove the effectiveness of our proposed text adversarial sample method.

2 Related Work

With the in-depth research on adversarial sample generation technology, the research has gradually covered the text field from the fields of image and audio. Because image data and audio data are continuous, text word data has the characteristics of discrete features, complex grammar rules, and abstract semantic forms. It is impossible to directly apply adversarial sample generation methods in the field of image and audio research to the field of natural language processing. Therefore, compared with continuous signal media such as images and sounds, the counterattacks of text data are more challenging. Zhang et al. pointed out that the difference between attacking image DNNs model and attacking a textual DNN model, which is compared in detail as shown in the following 3 points [7]:

(1) Continuous VS Discrete

The image data is continuous, and it is easy to be encoded as a numerical vector. The preprocessing operation is linear. The *lp norm* is usually used to measure the distance between the original sample and

the adversarial sample. Text data is so symbolic, discrete and non-linear that cannot be dealt with by preprocessing operations. it is difficult to define the disturbance on the text and measure the difference before and after the text sequence is changed. Therefore, for text confrontation, one suggestion is that the proposed models must carefully design variables or distance measurements for text disturbances, and the other suggestion is that the proposed models firstly map the text data to continuous data and then adopt image adversarial attack method.

(2) Unperceivable VS Perceivable

Small changes in image pixels are usually not easily perceivable by humans. Therefore, the adversarial samples generated in the image will not change the judgment of humans but only interfere with the judgment of the DNN model. However, small changes in the text, such as changes in characters or words, are easy to detect, which may cause the attack to fail. Of course, for the perturbation on texts of information transmission, people automatically ignore this change and can understand the meaning of the information.

(3) Semantic-Less VS Semantic

Small changes usually do not change the semantics of the image when the changes are trivial and unperceivable in the image. In contrast, the perturbation on texts would easily change the semantics of a word and a sentence.

In the process of generating adversarial samples by perturbing the text, according to the text granularity when adding perturbation, text perturbation strategies can be divided into three level attack: character level, word level and sentence level.

1) Character Level

Character level attack performs perturbation on the characters of the original text. Common methods include adding, deleting, replacing and swapping characters. Among them, for character replacement, there are random replacement [3], word replacement based on One-Hot coding [8], and replacement based on similar glyphs [9].

Gao et al. [6] proposed a character-level text adversarial sample generation scheme under the black box attack. First, the important words in the sentence are determined by the traversal method, and then the characters in the important words are disturbed by several substitution methods. but the generation speed is slower.

2) Word Level

The disturbance of words in the original text is mainly realized by replacing words. There are a variety of alternatives, including word vector similarity [10,11], near-synonyms, semantic words [12], spelling errors, synonyms [13], language model score [14], and other words replacement methods, which need to establish the corresponding thesaurus in advance. In addition to replacement, it also includes adding or deleting words [15]. Addition and deletion may affect the grammaticality and smoothness of the generated adversarial samples.

3) Sentence Level

Sentence level attack disturbs the entire sentence of the original text. Common sentence-level attack methods include paraphrasing [16,17], re-decoding after encoding [18], and adding irrelevant sentences [19]. Among them, Zhao [18] proposed to first use an inverter to map the original data into the vector space, search in the dense vector space corresponding to the data, add disturbance to get the adversarial samples; then use GAN as the generator to map the adversarial samples in the vector space back to the original data type.

Belinkov et al. [20] caused a decrease in the performance of machine learning using neural networks by adding, deleting, and exchanging text, which proved that the adversarial samples formed by the perturbation operation of the text interfered with the target system. Papernot et al. [21] directly used the FGSM method to perturb the word vector and then search for the word, corresponding to the closest word vector in the word vector space, to replace the original word.

Compared with the above three different granularity disturbances, the statement level disturbance often makes a huge difference between the adversarial sample and the original input data, which makes it difficult to control the quality of the generated adversarial sample and cannot guarantee effectiveness. The quality of the adversarial samples generated by character level disturbance is often very poor. The grammaticality may be destroyed and make sentence unreadable. Word level attack has better performance in terms of sample quality and attack success rate. In terms of the quality and effectiveness of adversarial samples, it is difficult to change the semantics of a sentence by the synonymous substitution of some words. Besides, the smoothness and fluency of the adversarial samples generated by using a language model are easier to be guaranteed.

This paper focuses on the generation technology of character-level adversarial samples and classifies the emotional attributes of sentences as test scenarios. By perturbing the characters, under the premise that there are not too many target sentences, the deep learning model which classify the emotional attributes of the target sentence will make the wrong classification result

3 Adversarial Sample Generation Model

This paper proposes an adversarial sample generation model. First, we use a long short-term memory model to determine the sentiment tendency of the sentence. All sentences for sentiment analysis are in English language. Then, we perform disturbing algorithm to find keywords in the sentence and select interference term to generate adversarial samples.

In terms of disturbance, the Jacobian matrix is used to sort the importance of all words in the sentence, keywords are found according to their priority, and the keywords are replaced, exchanged, deleted, and other disturbances. After generating several wrong words, according to the change of confidence, we choose the wrong word that has changed most, and then replace the keywords with the best wrong words to get a new text to achieve the effect of disturbance.

3.1 LSTM Model Introduction and Adversarial Text Generation

LSTM, long short-term memory, is a special RNN and mainly solves the problem of gradient disappearance and gradient explosion in the training process of long sequences. LSTM solves the long-dependency problem and was introduced by Hochreiter et al. [22]. Therefore, many researchers have improved and popularized it. Compared with ordinary RNN, LSTM can perform better in longer sequences.

There are three main stages inside LSTM:

1. Forget stage. This stage is mainly to selectively forget the input from the previous node. At this stage, the neural network will forget the unimportant content and leave the important content behind. The calculated data is used as the forget gate to control the input of the previous state, furthermore, determines which needs to be left and which needs to be forgotten.

2. Select the memory stage. This stage mainly selects and memorizes the inputs to ensure that those inputs are selectively “memorized”. The important content is mainly recorded, and the unimportant data content is less recorded. Based on these, it also selects the result of gate control signal to get the data content transmitted to the next state.

3. The output stage. This stage will determine which states will be regarded as the current output. It also scales the data content obtained in the previous stage which is changed through an activation function.

LSTM controls the transmission state through the gate state, remembering the long-term memory, and forgetting the unimportant information. Unlike ordinary RNN, there is only one memory stacking method. LSTM is very suitable for many tasks to require “long-term memory”. In many cases, the LSTM has been a great success and has been widely used. LSTM can effectively avoid RNN gradient explosion and other problems.

Firstly, we build the LSTM model, then, load the word vector data and fill the word vector matrix to map all the review data into numbers. next, we load the features and labels to map all the features into

numbers, and split them into the validation set and test set. Finally, we train the model and verify the model to generate adversarial text.

By using the IMDB data set, we build an LSTM model. Sentiment analysis of the sentence through LSTM, if it is different from the original sentence, the disturbance is successful.

3.2 Finding Important Keywords

To make the modified words have more influence on the original data, we use the following algorithm to select the important words in the sentence and determine whether they meet the replacement criteria through some restrictions.

Firstly, we sort the importance of each word to find the most important words as keywords, and use the matrix to judge the importance of each word in the x sentences (lines 2–4 as below). Then we sort each word in the x sentences by the importance obtained above (line 5 as below) and generates the corresponding disturbance. Finally, If the visual similarity between the new sentence and the original sentence is lower than e, the keyword selection fails, otherwise, the new sentence is returned (lines 6–14 as below).

Algorithm 1 Text words replace

Input: sentence x and its label y, classifier $F()$, threshould e

1:Inititialize: $x0 \leftarrow x$

2:for word xi in x do

3: To calculate axi

4:end for

5:$Torder \leftarrow Sort(x1, x2, x3,..., xn)$ according to axi ;

6:for xi in $Torder$ do

7: $word = SelectWord(xi,x0,y,F());$

8: $x0 \leftarrow$ replace the key words in the sentence with xi

9: if $S(x,x0) \leqslant e$ then

10: Return fail.

11: else

12: Return $x0$.

13: end if

14:end for

15:return None

Output: Changed sentence xchange

3.3 Selection of the Optimal Interference Term

There are five ways to generate interference:

(1) Insert.

(2) Delete.

(3) Exchange: randomly exchange two adjacent letters in a word, but do not change the first or last letter.

(4) Change the visually similar letters, such as 0 and o.

(5) In the context-aware word vector space, replace words with nearest neighbors.

Through the optimal interference term algorithm, we find the most suitable interference term and replace the corresponding keywords in the original sentence. The *produce* function generates disturbance

characters corresponding to the keyword *t*, and stores these different disturbance characters in words (line 2). We replace each of the different disturbing words with the keywords in the original sentence and use the *S* function to calculate the difference between the original sentence and the sentence after the replacement (lines 3–6). The disturbance with the largest difference is assigned to the word (line 7).

Algorithm 2 Select word

Input: sentence *x* and its label *y*

1:function *SelectWord(t,x,y,S())*

2: *words = produce(t)*;

3: for *b* in *words* do

4: *dates(p)* = replace *t* with *b* in *x*;

5: *score(p) = S(x) − S(dates(p))*;

6: end for

7: *word_choose* = arg max *score(p)*;

8: return *word_choose*;

9:end function

Output: *word_choose*

4. Experiment Analysis

4.1 Data Set

We used the IMDB data set, which can be used for text sentiment analysis. The data set contains 50,000 positive and negative film reviews about movies. The length of each piece of data is about 200 words. We use 25,000 pieces of data as the training set of the sentiment analysis model and the other 25,000 pieces of data as the test set. Part of the data set is shown in Tab. 1 below:

Table 1: Partial IMDB data set

IDX	Type	Comment	Tag
1	Test	Once again Mr.Costner has dragged out a movie for far…	Neg
12500	Test	I went and saw this movie last night after being coaxed to…	Pos
25000	Train	Story of a man who has unnatural feelings for a pig.Starts…	Neg
37500	Train	Bromwell High is a cartoon comedy. It ran at the same…	pos

4.2 Evaluation Standard

We perform sentiment analysis on the test set data, which is expressed as a floating-point number from 0 to 1. The value less than 0.5 represents negative while the value greater than 0.5 represents positive. We perturb the keywords in the sentence based on our proposed model. if the sentiment analysis of the sentence after the disturbance is the same as before the disturbance, the disturbance attack is unsuccessful. Alternatively, if the sentiment analysis is different, the disturbance attack is successful.

4.3 Experimental Results

After the test data sample generates disturbances, the sentiment analysis has different results from positive to negative. It can be seen that the text disturbance attack was successful. See Tab. 2 for details:

Table 2: Positive emotions to negative emotions

	Original	Counter
Test data	this is an early one from the boys but some people may not be satisfied with this one like all the others i found it to be different somehow than the your average stooge slapstick it was more funny for its jokes rather than the poke in the eye or slap watch for a hilarious part when larry grabs the stethoscope from moe and sings into it moe gives him a good smack that part made me crack up for a good ten minutes another hit for the	this is an early one from the boys but some people may not be satisfied with this one like all the others i found it to be different somehow than the your averages stooge slapstick it was worse funny for its joes rather than the poke in the eye or slap watch for a hilarious part when larry grabs the stethoscope from moe and sings into it moe gives him a good smack that part made me crack up for a good ten minutes another hit for the
Words	Average more more	Averages worse joes
Sentiment analysis scores	0.9059355	0.42782927
Emotional analysis results	positive	negative

In the above Tab. 2, we have made an emotional analysis of a sentence in the data set which was positive. By adding and replacing keywords in the test data, we have made another emotional analysis of the changed result. the result was negative.

Table 3: Negative emotions to positive emotions

Test data	Original	Counter
	Barbr unk first television spccal was simply fantastic from her skit as a child to her medley of songs in a unic department store everything was topnotch it was easy to understand how this special received awards not muddled down by guest appearances the focus remained on barhra thoughout the entire	Barbr unk first television spccal was simply vampire from her skit as a child to her medley of performances in a unic department store everything was topnotch it was easy to understand how this special received awards not muddled down by guest appearances the focus remained on barhra thou ghout the entire
Words	songs fantastic thoughout	performances vampire thou ghout
Sentiment analysis scores	0.4806286	0.5195926
Emotional analysis results	negative	positive

Similarly, in Tab. 3, we have made an emotional analysis of a sentence in the data set which is negative. By performing operations such as segmentation and replacement of keywords in the test data, we have made another emotional analysis of the changed result. the result was positive.

Our proposed adversarial sample generation model to modify keywords to achieve the success of text disturbance, and the success rate is high. The experimental result is shown in Tab. 4 below:

Table 4: Accuracy of adversarial text attacks

Dataset	Precision
IMDB	82.625%

5 Conclusion

Nowadays, people are relying more and more on words to express their thoughts. Words are everywhere on the Internet and social software. but in some cases, when words and phrases in the text are replaced, we do not know or notice that the sentences are not consistent with the meaning of the original text. These changes can deceive and disturb the classification judgment of artificial intelligence system. In order to defend this threat, how to generate adversarial samples for artificial intelligence system to improve detection capability is a very important aim. We proposed an adversarial sample generation model. Via finding important keywords and using five interference strategies, our model can make the emotion of the sentences to change for artificial intelligence systems without influencing people to read and understand. The experiment has proved our model effective.

Funding Statement: This work was supported by the National Key Research and Development Plan (Grant Nos. 2018YFB1800302 and 2019YFA0706404), the Natural Science Foundation of China (Grant No. 61702013), Joint of Beijing Natural Science Foundation and Education Commission (Grant No. KZ201810009011), Beijing Natural Science Foundation (Grant Nos. 4202020, 19L2021), Science and Technology Innovation Project of North China University of Technology (Grant No. 19XN108).

Conflicts of Interest: The authors declare that they have no conflicts of interest to report regarding the present study.

References

[1] J. Qiu, Z. Tian, C. Du, Q. Zuo, S. Su *et al.,* "A survey on access control in the age of Internet of Things", *IEEE Internet of Things Journal*. vol. 7, no. 6, pp. 4682–4696, 2020.

[2] C. Du, S. Liu, L. Si, Y. Guo and T. Jin, "Using object detection network for malware detection and identification in network traffic packets," *Computers, Materials & Continua*, vol. 64, no. 3, pp. 1785–1796, 2020.

[3] C. Szegedy, W. Zaremba, I. Sutskever, J. Bruna, D. Erhan *et al,* "Intriguing properties of neural networks," in *Proc. ICLR*, Banff, Canada, pp. 1–10, 2014.

[4] A. Ilyas, S. Santurkar, D. Tsipras, L. Engstrom, B. Tran *et al,* "Adversarial examples are not bugs, they are features," in *Proc. NeurIPS*, Vancouver, Canada, pp. 125–136, 2019.

[5] J. Su, D. V. Vargas, K. Sakurai, "One pixel attack for fooling deep neural networks," *IEEE Transactions on Evolutionary Computation*, vol. 23, no. 5, pp. 828–841, 2019.

[6] J. Gao, J. Lanchantin, M. L. Soffa and Y. Qi, "Black-box generation of adversarial text sequences to evade deep learning classifiers," in *Proc. SPW*, San Francisco, Ca, USA, pp. 50–56, 2018.

[7] W. E. Zhang, Q. Z. Sheng, A. Alhazmi and C. Li, "Adversarial attacks on deep learning models in natural language processing: a survey", *ACM Transactions on Intelligent Systems and Technology*, vol. 11, no. 3, pp. 1–40, 2020.

[8] J. Ebrahimi, A. Rao, D. Lowd and D. Dou, "HotFlip: White-box adversarial examples for text classification," in *Proc. ACL*, Melbourne, Australia, pp. 31–36, 2018.

[9] S. Eger, G. G. Şahin, A. Rücklé, J. Lee, C. Schulz *et al.,* "Text processing like humans do: visually attacking and shielding NLP systems," in *Proc. NAACL-HLT*, Minneapolis, Minnesota, pp. 1634–1647, 2019.

[10] M. Alzantot, Y. Sharma, A. Elgohary, B. Ho, M. Srivastava *et al.,* "Generating natural language adversarial examples," in *Proc. EMNLP*, Brussels, Belgium, pp. 2890–2896, 2018.

[11] D. Jin, Z. Jin, J. T. Zhou and P. Szolovits, "Is BERT really robust? A strong baseline for natural language attack on text classification and entailment," in *Proc. AAAI-20*, New York, USA, pp. 1765–1773, 2020.

[12] S. Ren, Y. Deng, K. He and W. Che, "Generating natural language adversarial examples through probability weighted word saliency," in *Proc. ACL*, Florence, Italia, pp. 1085–1097, 2019

[13] Y. Zang, F. Qi, C. Yang, Z. Liu, M. Zhang *et al.,* "Word-level textual adversarial attacking as combinatorial optimization," in *Proc. ACL*, Seattle, USA, pp. 6066–6080, 2020.

[14] H. Zhang, H. Zhou, N. Miao and L. Li, "Generating fluent adversarial examples for natural languages," in *Proc. ACL*, Florence, Italia, pp. 5564–5569, 2019.

[15] B. Liang, H. Li, M. Su, P. Bian, X. Li *et al.,* "Deep text classification can be fooled," in *Proc. IJCAI*, Stockholm, Sweden, pp. 4208–4215, 2018.

[16] M. T. Ribeiro, S. Singh and C. Guestrin, "Semantically equivalent adversarial rules for debugging NLP models," in *Proc. ACL*, Melbourne, Australia, pp. 856–865, 2018.

[17] M. Iyyer, J. Wieting, K. Gimpel and L. Zettlemoyer, "Adversarial example generation with syntactically controlled paraphrase networks," in *Proc. NAACL-HLT*, New Orleans, USA, pp. 1875–1885, 2018.

[18] Z. Zhao, D. Dua and S. Singh, "Generating natural adversarial examples," in *Proc. ICLR*, Vancouver, Canada, 2018.

[19] R. Jia and P. Liang, "Adversarial examples for evaluating reading comprehension systems," in *Proc. EMNLP*, Copenhagen, Denmark, pp. 2021–2031, 2017.

[20] Y. Belinkov and Y. Bisk, "Synthetic and natural noise both break neural machine translation", in *Proc. ICLR*, Vancouver, Canada, 2018.

[21] N. Papernot, P. McDaniel, A. Swami and R. Harang, "Crafting adversarial input sequences for recurrent neural networks," in *Proc. MILCOM*, 2016.

[22] S. Hochreiter and J. Schmidhuber, "Long short-term memory", *Neural Computation*, vol 9, no. 8, pp. 1735–1780, 1997.