Journal of Network Intelligence ©2024 ISSN 2414-8105 (Online)
Taiwan Ubiquitous Information Volume **9**, Number **1**, February **2024**

The Application of Adversarial Training Based on Gradient Constraint Optimization Method to Sentiment Analysis

Zhichun Xie

School of Data and Computer Science
Xiamen Institute of Technology, Xiamen 361021, China
66742494@qq.com

Jianhua Liu*, Renyuan Hu, Jiacan Wang

School of Computer Science and Mathematics
Fujian University of Technology, Fuzhou 350118, China
jhliu@fjnu.edu.cn, hhhry@foxmail.com, 1291102959@qq.com

Xiaofeng Wang

School of Data and Computer Science
Xiamen Institute of Technology, Xiamen 361021, China
1264613915@qq.com

*Corresponding author: Jianhua Liu

Received August 5, 2023, revised September 20, 2023, accepted November 29, 2023.

Abstract. *The previous adversarial training models failed to pay attention to the influence of the changing gradient of the loss function in the current training on the model. The perturbation injected into the model is only processed by standardization and constraints. This paper proposes a two-way encoding of a converter based on a fast gradient descent of a constraint strategy. The device characterizes the technical model and applies it to sentiment analysis tasks. The model can adapt to gradient changes, and can improve the model's convergence ability in the early stage of training. In the later stage of training, the perturbation can be appropriately increased according to the gradient to improve the robustness of the model. The experimental results show that even in the expectations of different languages, the classification accuracy of the model is significantly improved compared with the previous transfer learning models and adversarial training.*

Keywords: Gradient constraint; Adversarial training; Pre-training model; Sentiment analysis

1. **Introduction.** Sentiment analysis refers to the automated analysis of text content by extracting users' evaluations and opinions towards a specific object. It is a kind of text classification problems, which can be addressed through traditional machine learning methods such as Support Vector Machines [1], Naive Bayes [2], Maximum Entropy [3], and K-Nearest Neighbor algorithms [4]. Traditional approaches mainly involve manual annotation of a part of the text data as a training set, followed by the manual extraction of text features and training of a machine learning classification model. The model is then used to predict the classification of unlabeled data and output the final predicted classification results. Traditional text classification methods are based on machine learning, and have achieved considerable success. However, they rely heavily on complex manual rules

for text feature engineering, and the appropriateness of feature engineering strategies can greatly affect the effectiveness of sentiment classification.

In recent years, Deep Learning has achieved remarkable results in the field of Natural Language Processing (NLP), and has become a current research hotspot [5]. Word2Vec [6] is currently widely used for training word vectors in NLP. In 2014, Pennington et al. [7] proposed the Glove model, which improves the training speed and model stability of word vectors on large corpus datasets and has since been widely applied. Through deep learning-based training of corpus data, word vectors can be pre-trained to form a pre-trained model. Therefore, a pre-trained model is a neural network architecture trained on a large amount of data sets and can be used for downstream task implementation. Pre-trained models generally outperform traditional neural networks in many NLP tasks. Through further in-depth research on pre-trained models, rich pre-trained models such as ELMo [8], Transformer [9], and Bidirectional Encoder Representations from Transformers (BERT) [10] have been successively proposed, among which BERT is currently the most widely used pre-trained model. At the same time, adversarial training has received widespread attention as a means to enhance model robustness. Currently, adversarial learning generally has two meanings in deep learning: one is Generative Adversarial Networks (GAN) [11], representing a large class of advanced generative models; the other is the field related to adversarial attacks and adversarial samples, which is related to GAN, but is quite different, and mainly concerned with the robustness of models under small perturbations. In previous studies, researchers found that Neural Networks can be easily deceived by samples with slight perturbations. Szegedy et al. [12] proposed the concept of adversarial samples by intentionally adding imperceptible perturbations to the input samples, which cause the model to give an incorrect output with high confidence. Goodfellow et al. [13] proposed a Fast Gradient Sign Method (FGSM) to reduce the disturbance caused by extreme non-linearity of deep neural networks. Their model design uses non-linear effects to iteratively resist adversarial perturbations and improve model performance. Madry et al. [14] proposed a Projected Gradient Descent (PGD) method to resist adversarial attacks, which proposed a Min-Max optimization objective to enhance model robustness. Miyato et al. [15] introduced adversarial and virtual adversarial training into text classification, applied perturbations to word vectors instead of the original input, and improved the model's classification ability through regularization. Goodfellow et al. [16] proposed a more concise and effective Fast Gradient Method (FGM) to increase the model's loss value and make it capable of handling slight perturbations. Dong et al. [17] combined the BERT model with adversarial training, which improved the model's score and was applied to named entity recognition tasks. Zhao et al. [18] proposed a mixed regularization adversarial training method for multi-level attacks to improve the model's inherent robustness, achieving good results in text classification tasks. Tariq A et al. [19] use Adversarial Training as a means of regularization for fake news classification. We train two transformed-based encoder models using adversarial examples that help the model learn noise invariant representations. Wang et al. [20] proposed dual adversarial networks that utilizes a domain-knowledge generator during adversarial training to produce domain-specific knowledge, and a domain discriminator to recognize the domain label of the produced knowledge. To address these issues, this paper proposes an adversarial training method with a gradient-constrained strategy (GCFGM). This method can incorporate gradients into perturbations, so that when the model is far from the optimal point, it can reduce the perturbation and help the model converge faster, while increasing the perturbation when the model is close to the optimal point. And it is combined with a pre-trained model to propose a pre-trained model based on the gradient-constrained strategy (BERT-GCFGM), which avoids the above shortcomings and improves the accuracy of model classification.

2. The Related Work.

2.1. **Transformer.** The encoder part of the Transformer model includes two submodules: a self-attention layer that uses multi-head attention and a fully connected feed-forward neural network, both of which perform data normalization operations. Each submodule in the model uses residual connections to address the problem of neural network degradation. Based on the Seq2Seq structure, the Transformer model changes the traditional Encoder-Decoder architecture, which relies on the RNN pattern, and is built solely using attention mechanisms and fully connected neural networks. The input data is processed through word embedding and position encoding, allowing the model to learn the positional relationship of the text sequence, and then learns the relationship between words in the text sequence through the multi-head attention mechanism. The structure of the Transformer model is shown in Figure 1.

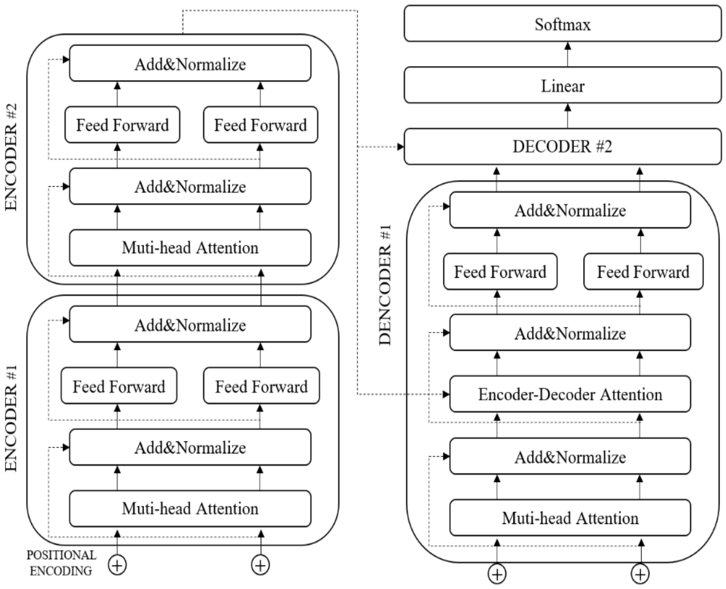

Figure 1. Transformer Model

The dashed arrow in the figure represents the residual connection, which is used to solve the difficulty of training deep neural networks. By passing the information from the previous layer to the next layer without any difference, the model can effectively focus only on the differential part. The calculation formulas for attention are shown in Equations (1) and (2), where Q, K, and V are initialized randomly and continuously updated through training, and they are the inputs to the attention layer. The multi-head attention mechanism combines multiple self-attention mechanisms, allowing the model to learn different aspects of the content through different heads, and giving the model greater capacity. It can help the model scale and avoid the softmax result being either 0 or 1. The matrix W^0 is also randomly initialized, and the attention matrices learned by each head are concatenated.

$$\dot{A}ttention(Q, K, V) = softmax(\frac{QK^T}{\sqrt{d_k}})V \tag{1}$$

$$MutiHead(Q, K, V) = Concat(head_1, \ldots, head_h)W^0 \tag{2}$$

2.2. **Bert Model.** Traditional word embedding tools like Word2Vec are based on shallow neural network models to provide word embeddings as features. In contrast, the BERT model can be integrated into downstream tasks and adjusted for specific task systems. BERT is based on a bidirectional Transformer model, primarily using the Encoder module for computation, as shown in Figure 2. It uses a masked language model (MLM) for modeling, allowing the output sequence to more comprehensively learn text information in different directions and provide better initial parameters for subsequent fine-tuning.

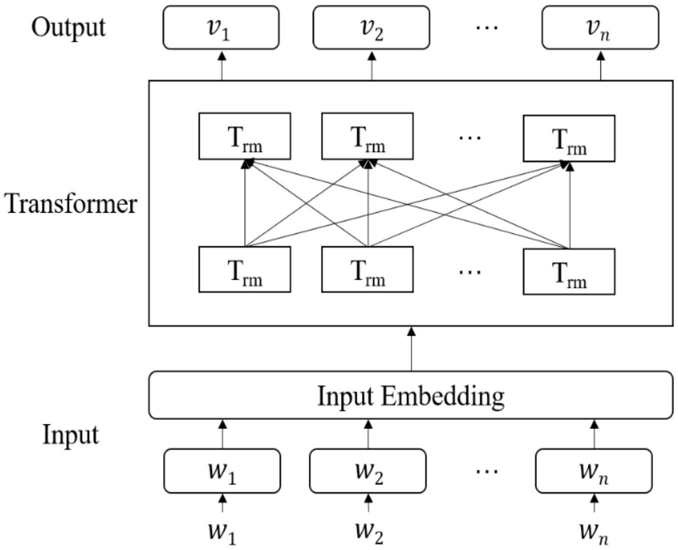

Figure 2. BERT model Structure Diagram

3. **An Adversarial Training Model Based on Gradient Constraint Strategy (BERT-GCFGM).** This paper proposes a transfer learning model based on a gradient-constrained strategy and a fast gradient descent algorithm. As shown in Figure 4, the BERT-GCFGM model adds perturbation r to the original model parameters to enhance the model's robustness when approaching the optimal solution, and reduce perturbation to help the model converge quickly when far from the optimal solution, ultimately improving the model's classification accuracy.

3.1. **The Input of the model.** First, stop words and special meaningless symbols are removed from the Chinese text data, and then a word dictionary is constructed by selecting the most common misspelled words in the field of sentiment analysis to perform misspelling replacement on the text sequence. For English text data, special meaningless symbols are removed and the text is converted to lowercase. Each input of the BERT model is a sum of token embeddings, segment embeddings, and position embeddings. Token embeddings are randomly initialized and their values are automatically learned during model training. They are used to capture the global semantic information of the text and are fused with the semantic information of single-word words. Segment embeddings are used to differentiate the context in which words appear. Position embeddings are added to distinguish between the semantic information carried by words appearing in different positions in the text (e.g., "I love you" vs. "You love me"). Therefore, the BERT model attaches a different vector to words in different positions to distinguish them, and the input format is shown in Figure 3.

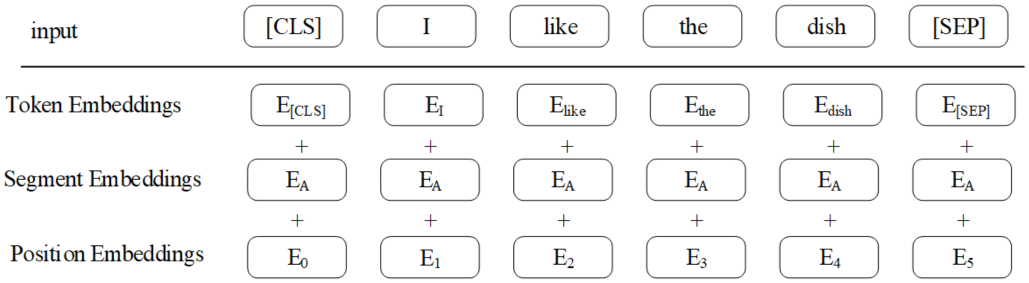

Figure 3. The input of the Model

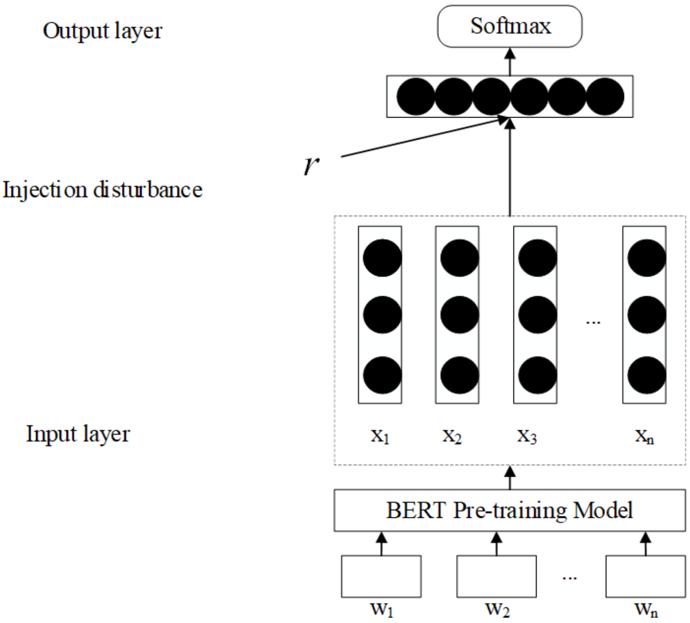

Figure 4. The Model of BERT-GCFGM

3.2. Gradient-Constrained Adversarial Training.

3.2.1. ***Min-Max Optimization Objective.*** In this paper, Min-Max is used as the objective function, as shown in Equation 3.

$$\min_{\theta} E_{(x,y)\sim D}[\max_{\Delta x\in\Omega} L(x+\Delta x, y; \theta)] \tag{3}$$

The objective is to maximize the internal loss function and minimize the external empirical risk. D represents the training set, x represents the original input, θ represents the model parameters, and $L(x,\ y;\ \theta)$ represents the loss value of a single sample, as shown in Equation (4). Δx represents the adversarial perturbation and Ω represents the perturbation space.

$$\log L(x + \Delta x, y; \theta) \tag{4}$$

Injecting Δx into the original input x is to make $L(x,\, y;\, \theta)$ as large as possible, causing the existing model to make incorrect predictions. Δx satisfies the constraint $\|\Delta x\| \leq \varepsilon$, where ε is a constant. After constructing an adversarial sample $x + \Delta x$ for each sample, $(x + \Delta x,\, y)$ is used as the data to minimize the loss value and update the parameter θ. This process is repeated to obtain the maximum perturbation and minimum gradient.

3.2.2. ***The Fast Gradient Descent Algorithm Based on Gradient Constraint Strategy.*** Since the *Min-Max* optimization objective requires increasing the value of $L(x,\, y;\, \theta)$, a gradient ascent method can be used to increase the loss value of the objective function. Therefore, Δx is taken as shown in Equation (5).

$$\Delta x = \varepsilon \nabla_x L(x, y; \theta) \tag{5}$$

To prevent Δx from becoming too large, it is usually normalized by applying Equation (6) to the text.

$$\Delta x = \varepsilon \frac{\nabla_x L(x, y; \theta)}{\|\nabla_x L(x, y; \theta)\|} \tag{6}$$

To incorporate the gradient factor into the perturbation, this paper proposes an optimization objective as shown in Equation (7).

$$\Delta x = \varepsilon (1 - sigmoid(\nabla_x L(x, y; \theta))) \frac{\nabla_x L(x, y; \theta)}{\|\nabla_x L(x, y; \theta)\|} \tag{7}$$

This objective function increases the perturbation when approaching the optimal solution and reduces it when far from the optimal solution to achieve rapid convergence. Finally, after simplification, Equation (8) is obtained.

$$\min_{\theta} E_{(x,y)\sim D}[L(x + \Delta x, y; \theta)] \tag{8}$$

4. Experimental Results and Analysis.

Three experimental methods were mainly used in this study: (1) Comparative experiments were conducted between the BERT word vector model and three other different word vector models, Word2Vec, Glove, and ELMo, to verify the superiority of the BERT word vector model. (2) Comparative experiments were conducted between the BERT-GCFGM model and other deep learning models for sentiment classification to verify the effectiveness of the proposed model in improving the efficiency of sentiment classification.

4.1. Experimental Settings.

4.1.1. ***Experimental Environment.*** The experimental environment in this study is as follows: Windows 10 operating system, Intel Core i5-8300H CPU, GeForce GTX 1060 6GB GPU, DDR4 16GB memory, TensorFlow 2.2.0-GPU development environment, and JetBrains Pycharm development tool.

4.1.2. ***Experimental Data.*** Two languages, Chinese and English, were used in the experiment, with each language including a binary classification data and a three-classification data. The training set and test set were independent datasets to ensure the effectiveness of the proposed method. For the Chinese text data, stop words and special meaningless symbols were removed, and a dictionary was constructed to replace commonly misspelled words in sentiment analysis. For the English text data, special meaningless symbols were removed and the text was converted to lowercase. The Chinese experimental data used the open-source data from Data Fountain, including the O2O food-related comments (abbreviated as O20) and public sentiment data during the epidemic (abbreviated as Cov19). The English experimental data included SST-2 and Twitter airline comment data, as shown in Table 1. "CN" represents Chinese text data, and "EN" represents English text data.

Table 1. The Statistical Results of Experimental Data

Language	DataSets	Positive Sentence	Negative Sentence	Neutral Sentence
CN	O2O-Train	6793	2417	0
	O2O-Test	1698	604	0
	Cov19-Train	20313	13521	46095
	Cov19-Test	5079	3381	11524
EN	SST2-Train	2788	2688	0
	SST2-Test	722	622	0
	Twitter-Train	1883	2465	7335
	Twitter-Test	471	617	1834

4.1.3. ***Evaluation Metrics.*** The evaluation metrics used in this study are precision, recall, and $F1$-score. Precision refers to the proportion of correctly predicted positive samples among all samples predicted as positive, while recall represents the proportion of correctly predicted positive samples among all positive samples. $F1$-score, a comprehensive measure of precision and recall, is used as one of the evaluation metrics for the model's classification results, as shown in Equations (9) to (11):

$$Precision = \frac{TP}{TP + FP} \tag{9}$$

$$Recall = \frac{TP}{TP + FN} \tag{10}$$

$$F1 = \frac{2PR}{P + R} \tag{11}$$

In this context, TP (True Positive) refers to the number of positive samples that are correctly classified as positive, FP (False Positive) refers to the number of negative samples that are incorrectly classified as positive, and FN (False Negative) refers to the number of positive samples that are incorrectly classified as negative.

4.1.4. ***Model Parameter Settings.*** Since the selection of model parameters can have a significant impact on the results, the experiment used the method of controlling variables. The BiLSTM hidden layer nodes were set to 64, 128, and 256, the Adam optimizer was used to optimize the function, and the convolutional layer padding mode was set to "same". The BERT model was selected using a mixed-language model to ensure the initial weights of the model were the same. Through multiple comparative experiments, it was found that the BERT-GCFGM classification model performed best when the parameters shown in Table 2 were used.

Table 2. Model Parameter Settings

Parameters	Value
BERT Model	Multilingual Cased
Optimizer	Adam
ε Disturbance	0.6
Learning Rate	2e-5
Batch_size	16
Dropout	0.3

4.2. **Word Embedding Model Comparison Experiment.**

4.2.1. ***Experiment Content and Method.*** In this section, we compare four different word embedding models, Word2Vec, Glove, ELMo, and BERT, for sentiment classification on four datasets. The purpose is to verify the rationale behind selecting the BERT model. The experimental results are shown in Table 3.

Table 3. Comparison Results Based on Different Word Embedding Models

DataSets	Models	P/%	R/%	F/%
CN-o2o	Word2Vec-GCFGM	92.83	82.55	87.18
	Glove-GCFGM	91.17	86.58	88.04
	ELMo-GCFGM	92.04	87.49	89.70
	BERT-GCFGM	94.17	97.33	95.16
CN-Cov19	Word2Vec-GCFGM	75.90	75.58	76.75
	Glove-GCFGM	77.92	77.21	78.44
	ELMo-GCFGM	84.94	85.64	84.33
	BERT-GCFGM	86.46	86.03	86.17
EN-SST2	Word2Vec-GCFGM	79.11	78.05	79.39
	Glove-GCFGM	79.28	77.78	80.17
	ELMo-GCFGM	87.72	83.89	84.96
	BERT-GCFGM	89.83	83.92	86.77
EN-Twitter	Word2Vec-GCFGM	77.89	77.21	78.44
	Glove-GCFGM	79.36	77.35	79.15
	ELMo-GCFGM	86.67	79.32	82.21
	BERT-GCFGM	84.94	85.64	84.33

4.2.2. ***Experimental Results and Analysis.*** As shown in Table 3, the Glove-GCFGM model performs better than Word2Vec-GCFGM because Glove utilizes co-occurrence information through matrix factorization, enabling it to learn global information while also focusing on context, thereby enhancing its semantic representation capabilities. ELMo and BERT can both dynamically represent word vectors and fine-tune their semantic representation capabilities based on downstream tasks, helping models learn domain-specific knowledge and improve the efficiency of identifying polysemous words. Additionally, the generated word vector features are more abundant, resulting in significant improvements in model scores. Further analysis of Table 3 indicates that compared to the Word2Vec model, ELMo achieved an average increase of 4.16%, 4.84%, and 4.50% in precision, recall, and $F1$-score, respectively, on the four datasets. Unlike ELMo, which uses LSTM for word vector feature extraction, BERT employs a more powerful Transformer encoder for sentiment representation, resulting in further improvements in feature extraction capabilities. Compared to the ELMo model, the BERT model achieved an increase in $F1$-score of 2.54%, 1.92%, and 2.22% on the four datasets, and all models using BERT as the word vector tool obtained the highest $F1$-scores.

4.3. Comparison Experiment of Classification Models.

4.3.1. ***Experiment Design and Method.*** To validate the effectiveness of the proposed BERT-GCFGM model, this section compares it with three typical neural network models and three recently proposed deep learning models based on BERT on four datasets. The seven experimental models are described below: CNN. A convolutional neural network model proposed in reference [21,22]. It uses independent sentences as the input of the network model, ignoring the temporal sequence of the text and long-distance dependencies between sentences. It is a basic convolutional network model. BiLSTM. A BiLSTM model proposed in reference [23]. This model can process time series, but it does not perform feature extraction on the input text sequence. Marginal information may interfere with the model's classification results, making it difficult to effectively identify the sentiment polarity of the sentence. BERT. A pre-training model based on transfer learning proposed in reference [10]. It uses the Encoder module of Transformer to construct the model, and combines multi-head attention mechanism and feed-forward neural network to learn input information. Compared with traditional neural networks, it has made significant breakthroughs. BERT-PGD. A PGD algorithm proposed in reference [14]. It uses the Min-Max optimization objective and maximizes the loss value through multiple iterations. If the model norm exceeds a certain value, it will be scaled down. This greatly improves the robustness of the model. BERT-FGM. An FGM algorithm proposed in reference [16]. It prevents excessive perturbation by normalizing gradients and effectively enhances the robustness of the model. BERT-GCFGM. The proposed model combines gradient-constrained adversarial training with BERT. The current gradient factor is added to the perturbation, which helps the model converge quickly in the early stage of training and can increase the perturbation to improve the robustness of the model in the later stage. It has shown significant improvements on all four different datasets.

4.3.2. ***Experimental Results and Analysis.*** Table 4 shows the results of different models on four datasets, where P, R, and F represent precision, recall, and $F1$ score, respectively.

According to the experimental results shown in Table 5, the classification performance of the BERT model based on the transfer learning idea is far superior to that of traditional neural network models. The average $F1$ scores of CNN and BiLSTM on the four datasets are only 76.76% and 78.31%, respectively, while the BERT model, compared with

Table 4. Results of different models on four data sets

Models	Index	Data Set			
		CN		EN	
		o2o	Cov19	SST2	Twitter
CNN	P/%	86.67	72.99	78.65	74.33
	R/%	79.32	67.87	77.94	75.58
	F/%	82.21	70.40	78.29	76.14
BiLSTM	P/%	92.83	74.93	79.86	77.41
	R/%	82.55	65.45	79.44	77.29
	F/%	87.18	69.86	79.64	78.38
BERT	P/%	92.04	83.80	83.81	82.18
	R/%	93.19	80.73	84.27	80.94
	F/%	92.61	82.75	82.74	81.66
BERT-PGD	P/%	92.77	84.59	83.32	83.31
	R/%	94.46	83.94	84.88	82.17
	F/%	93.60	84.26	84.09	82.73
BERT-FGM	P/%	93.06	85.61	84.73	82.12
	R/%	95.33	82.01	84.60	82.52
	F/%	94.18	83.77	84.66	82.32
BERT-GCFGM	P/%	94.17	86.46	85.75	84.90
	R/%	97.33	86.03	88.64	83.64
	F/%	95.16	86.17	86.58	84.25

*Note: P, R and F represent accuracy rate, recall rate and F1

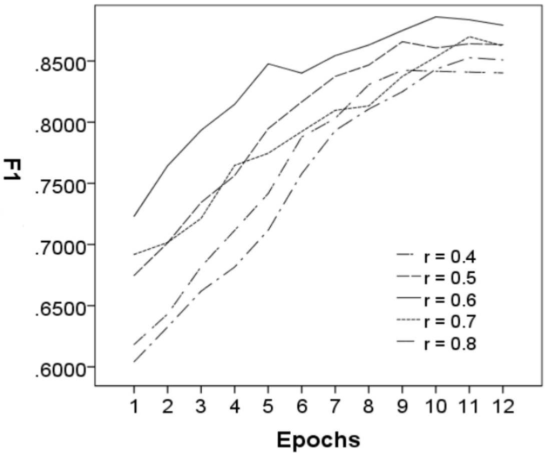

Figure 5. The convergence of models under different disturbance coefficients

traditional neural networks, has an overall improvement of 7.86% and 5.88% in terms of average $F1$ scores on the four datasets. The proposed BERT-GCFGM model achieves better classification performance than other network models on the four datasets. Compared with the network structure using the BERT model, the proposed BERT-GCFGM model improves the average $F1$ score on the four datasets by 3.63%, and the performance on the best-performing Cov19 dataset is improved by 4.16%. Compared with recent PGD and FGM adversarial training algorithms, the $F1$ score is improved by an average of 2.53% and 2.07%, respectively, which verifies the effectiveness of the proposed method in this paper. At the same time, in order to verify the impact of different initial perturbation coefficients on the model, this paper also conducted ablation experiments with perturbation coefficients of 0.4-0.8, to test the classification performance of the model under different perturbation coefficients. The experiment found that the model has the best classification performance when the perturbation coefficient is set to 0.6.

5. **Conclusion.** In conclusion, this paper proposes an adversarial training method with a gradient constraint strategy to address the problem of the impact of the changing loss function gradients during each training process on the model. Through comparisons with traditional neural network models and recent adversarial training methods in sentiment classification task experiments, the effectiveness of the BERT-GCFGM model is verified. The experimental content of this paper is limited to binary and ternary classification problems, and in the future, further research is needed to investigate the effectiveness of this model for more nuanced sentiment polarity problems.

Acknowledgements. The research is supported by: Xiamen Municipal Natural Science Foudation,china(grant number: 3502Z20227332), Fujian Provincal Natural Science Foundation,china(grant number: 2023J01349).

REFERENCES

[1] S. Styawati, A. Nurkholis, AA. Aldino, S. Samsugi, E. Suryati, RP. Cahyono, "Sentiment analysis on online transportation reviews using Word2Vec text embedding model feature extraction and support vector machine (SVM) algorithm[C]," in *International Seminar on Machine Learning, Optimization, and Data Science (ISMODE 2021).* IEEE, 2022, pp.163-167.

[2] PPM. Surya, B. Subbulakshmi, "Sentimental analysis using Naive Bayes classifier," in *International Conference on Vision Towards Emerging Trends in Communication and Networking (ViTECoN 2019).* IEEE, 2019, pp.1-5.

[3] X. Xie, S. Ge, F. Hu, M. Xie, N. Jiang, "An improved algorithm for sentiment analysis based on maximum entropy," *Soft Computing*, vol.23, pp.599-611, 2019

[4] O. Kaminska, C. Cornelis, V. Hoste, "Fuzzy Rough Nearest Neighbour Methods for Aspect-Based Sentiment Analysis," *Electronics*, vol.12, no.5, 1088, 2023

[5] L. Zhang, S. Wang, B. Liu, "Deep learning for sentiment analysis: A survey," *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery,* vol.8, no.4: e1253, 2018

[6] T. Mikolov, I. Sutskever, K. Chen, GS. Corrado, J. Dean, "Distributed representations of words and phrases and their compositionality," *Advances in Neural Information Processing Systems(NIPS 2013).* 2013,pp. 3111-3119.

[7] J. Pennington, R. Socher, CD. Manning, "Glove: Global vectors for word representation," in *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP).* 2014, pp.1532-1543.

[8] ME. Peters, M. Neumann, M. Iyyer, et al. Deep contextualized word representations *arXiv preprint.* arXiv:1802.05365, 2018.

[9] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, AN. Gomez, Ł. Kaiser, I. Polosukhin, "Attention is all you need," in *Advances in Neural Information Processing Systems(NIPS 2017).* 2017, pp.5998-6008.

[10] J. Devlin, MW. Chang, K. Lee, K. Toutanova, "BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding," *arXiv preprint.* arXiv:1810.04805, 2018.

[11] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, Y. Bengio, "Generative adversarial nets," in *Proceedings of the 27th International Conference on Neural Information Processing Systems(NIPS 2014).* 2014, Vol.2, pp.2672-2680.

[12] C. Szegedy, W. Zaremba, I. Sutskever, J. Bruna, D. Erhan, I. Goodfellow, R. Fergus, "Intriguing properties of neural networks," *arXiv preprint.* arXiv:1312.6199, 2013.

[13] T. Miyato, AM. Dai, I. Goodfellow, "Adversarial training methods for semi-supervised text classification," *arXiv preprint.* arXiv:1605.07725, 2016.

[14] A. Madry, A. Makelov, L. Schmidt, D. Tsipras, A. Vladu, "Towards deep learning models resistant to adversarial attacks," *arXiv preprint.* arXiv:1706.06083, 2017.

[15] T. Miyato, S. Maeda, M. Koyama, S. Ishii, "Virtual adversarial training: a regularization method for supervised and semi-supervised learning," *IEEE Transactions on Pattern Analysis and Machine Intelligence.* Vol.41, No.8, 2018, pp.1979-1993.

[16] T. Miyato, AM. Dai, I. Goodfellow, "Adversarial training methods for semi-supervised text classification," *arXiv preprint.* arXiv:1605.07725, 2016.

[17] Z. Dong, R.-Q. Shao, Y.-L. Chen, J.-W. Chen, "Named entity recognition in the food field based on BERT and Adversarial training," in *2021 33rd Chinese Control and Decision Conference (CCDC).* IEEE, 2021, pp.2219-2226.

[18] J. Zhao, P. Wei, W. Mao, "Robust Neural Text Classification and Entailment via Mixup Regularized Adversarial Training," in *Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval* . 2021, pp.1778-1782.

[19] A. Tariq, A. Mehmood, M. Elhadef, MUG. Khan, "Adversarial Training for Fake News Classification," *IEEE Access.* Vol.2022, No.10, pp.82706-82715, 2022.

[20] X. Wang, Y. Du, D. Chen, X. Li, X. Chen, Y. Fan, C. Xie, Y. Li, J. Liu, H. Li, "Dual adversarial network with meta-learning for domain-generalized few-shot text classification," *Applied Soft Computing,* vol.146, 110697, 2023.

[21] Y. Kim, "Convolutional neural networks for sentence classification," *arXiv preprint.* arXiv:1408.5882, 2014.

[22] Y. Dai, L. Shou, M. Gong, X. Xia, Z. Kang, Z. Xu, D. Jiang, "Graph fusion network for text classification," *Knowledge-based Systems,* vol.236, 107659, 2022.

[23] P. Zhou, W. Shi, J. Tian, Z. Qi, B Li, H. Hao, B. Xu, "Attention-based bidirectional long short-term memory networks for relation classification," in *Proceedings of the 54th Annual MMeeting of the Association for Computational Linguistics (volume 2: Short papers).* 2016, pp.207-212.