Journal of Combinatorial Mathematics
and Combinatorial Computing
www.combinatorialpress.com/jcmcc

Combinatorial Press

Research on Generating and Optimizing English Writing Style Based on Generative Adversarial Networks

Xiao Liu[1,] ✉

[1] *Department of Basic Science, Shaanxi University of International Trade & Commerce, Xi'an, Shaanxi, 712046, China*

ABSTRACT

In this paper, we design and implement a model network for English writing style generation using UNet network as well as ViT for encoding and decoding, and PatchGAN to enhance the identification speed. Based on the CRF-NLG model to identify and extract professional English terms, and design a special loss function to optimize the quality of writing style generation. The F1 value is used to evaluate the model recognition ability, and the writing style generation effect is explored by controlled experiments of the proposed model and three baseline models. The practical application results of the proposed model are visualized from four perspectives: overall evaluation, style strength, content preservation, and fluency, to verify its practical application effect. The results show that the proposed model exhibits the strongest performance in the two levels of content preservation and fluency, which are improved by 12.71% and 39.11%, respectively, compared with the existing GAN-based style generation model. Of the 119 modifications 92 (77.3%) were better, 17 (14.3%) were average, and only 11 (9.2%) were worse.

Keywords: writing style generation, CRF-NLG, PatchGAN, loss function

1. Introduction

As artificial intelligence continues to evolve, it has become a transformative technology that has the potential to revolutionize various industries, including the field of writing. AI-powered writing tools offer a range of features that can help writers create high-quality content efficiently [1-4]. One of the main advantages of AI in English writing is its ability to provide with a certain writing style and provide real-time feedback and suggestions to creators, AI-powered writing assistants can analyze the text and identify areas for improvement, such as grammar, clarity, and style [5-8]. These advantages of AI are invaluable for writers who want to improve the readability and effectiveness of their writing, especially the application of Generative Adversarial Networks to English writing [9-11].

Generative Adversarial Network (GAN) is a deep learning model in the field of Artificial Intelligence,

✉ Corresponding author.
E-mail address: liuxiao001485@163.com (X. Liu).
Received 25 October 2024; Revised 10 December 2024; Accepted 05 March 2025; Published Online 16 April 2025.
DOI: 10.61091/jcmcc127b-150

which consists of two parts: a generator and a discriminator. The generator is responsible for generating data samples, while the discriminator is responsible for determining whether the generated samples are real samples or fake samples generated by the generator [12-15]. During the training process, the generator and the discriminator continuously optimize themselves by means of confrontation, which ultimately makes the generator able to generate samples similar to the real samples, and by using it in English writing, it can generate writing styles and play a role in optimization [16-19]. However, there are many problems in the training process of generative adversarial network, such as pattern collapse and pattern collapse, etc. With the continuous development of deep learning technology, the optimization strategy of generative model structure of generative adversarial network will be further improved and perfected [20-23].

Literature [24] describes the current situation and problems of text complexity research based on deep learning, and based on the BPNN algorithm, the text complexity of Chinese and foreign academic English writing is analyzed, revealing that the BPNN algorithm can effectively analyze the text complexity of academic English writing. Literature [25] outlines the application strategy of automatic evaluation system in college English writing and teaching to better realize the integration of information technology and subject teaching and improve students' English writing ability and level. Literature [26] proposed MsCAEWL, a computer-assisted English writing learning system, which meets the requirements of writing feedback theory and has a significant impact on improving students' English writing. Literature [27] constructed an error correction model for English writing errors and applied it to the actual system, thus realizing the automatic checking and error correction of English composition writing errors and enabling students to achieve the purpose of independent practice. Literature [28] proposes a text vector calculation method, which shows more convenient, fast, concise and effective features compared with traditional manual scoring, and is of great significance to improve the efficiency of college English writing teaching. Literature [29] proposed the concept of integrating artificial intelligence and deep learning technology into English mobile learning platform, revealing that the method can optimize the existing platform and provide more ideas for the development of online English learning, which has good theoretical and practical value. Literature [30] analyzes the teaching of college English writing under deep learning, including the teaching design of several segments before, during, and after class, as well as the enhancement of deep learning through the use of personal, peer, and instructor feedback, so as to achieve the improvement of students' writing ability. Literature [31] proposed a text-based interface for mobile platforms based on deep learning, gave an interface-based application for teaching English, and created an English teaching application utilizing the interface, which effectively facilitated the learning of alphabetic writing and handwritten words. Literature [32] launched a study from the aspects of English vocabulary, reading comprehension, and composition, explored the relationship between the quality of teaching and the knowledge points of English, and pointed out that by testing the model performance of the depth of English vocabulary, reading comprehension, and writing and other knowledge points, it can be a good way to evaluate and analyze the students' mastery of the English learning. Literature [33] introduced an English assisted learning system based on deep learning, which can assess and analyze students' language skills in vocabulary, pronunciation, and other aspects, and provide personalized teaching content and suggestions, which meets students' English learning needs and provides personalized learning suggestions and learning resources. The above study examines the application of deep learning in English language teaching, including its positive role in applied writing, English proficiency assessment, etc., and also reveals the current wide application of artificial intelligence in education.

In this paper, based on Lafferty's Conditional Random Field model, a CRF-NLG model embedded in natural language grammar is designed to recognize and extract professional English terms. A combination of UNet network and ViT is used to form a generator, and PatchGAN is used as a model discriminator. The optimization method of generative adversarial network is analyzed from five levels of loss function, and the English writing style generation model based on generative adversarial

network is proposed. The level of writing style recognition of the proposed model is evaluated by comparing the accuracy, recall and F1 value of the model. Comparison experiments are conducted between the proposed model and three baseline models to explore the superiority of the proposed model's style generation in terms of three dimensions: style strength, content preservation, and fluency. The proposed model is put into field user experiments to assess the effectiveness of the model in assisting real-world writing tasks through expert evaluation.

2. Professional English terminology identification and extraction

The corpus analysis center will classify and analyze the processing according to the task categories set by the knowledge base manager, and its specific flow is shown in Figure 1. The corpus analysis center assigns raw corpus of different natures to different specialized sorting centers, from which more accurate information can be obtained. Some highly formatted raw webpage information can even be extracted directly with only simple format analysis, without the need for complex mathematical models, e.g., word frequency collection. Some of them only need a simple mathematical model to obtain high recall and accuracy. Some of them need complex mathematical models, even with other technical methods, such as: professional vocabulary discovery, proper noun recognition, lexical annotation and so on.

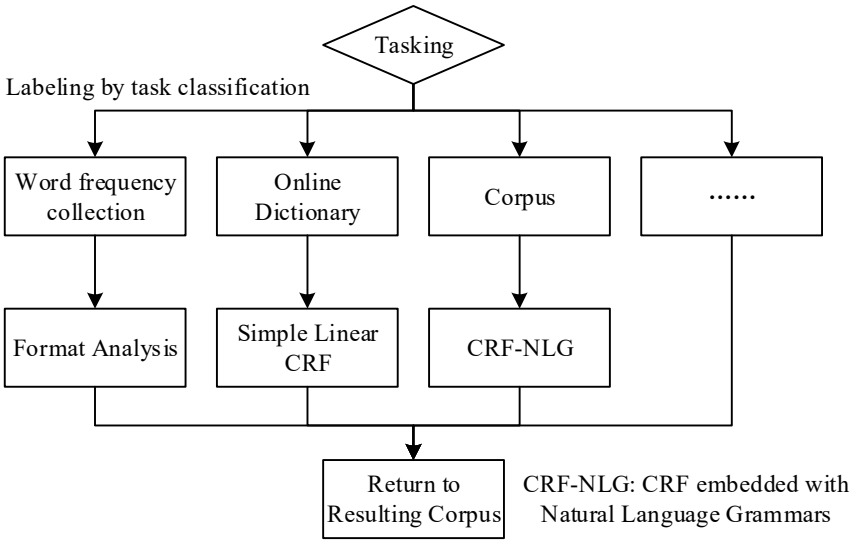


Fig. 1. Processing model of corpus analysis center

In this paper, we make appropriate additions to Lafferty's Conditional Random Field model and design the CRF-NLG model embedded in Natural Language Grammar. Conditional Random Field (CRF) is a discriminative probabilistic graph model proposed on the basis of Maximum Entropy Model and Hidden Markov Model. Bayesian Networks, Markov Random Fields, and Hidden Markov Models are generative models. Support vector machines, maximum entropy Markov models and conditional random fields are discriminative models. Hidden Markov models require three independence conditions, i.e., the current state is determined only by the previous state; the probability of an observation, is related only to the current implicit state: the state is time-independent. For simple problems, this assumption still seems reasonable. However, the formation of most real observation sequences in the real world is constrained by multiple interacting features, and by interdependence between a larger range of elements in the observation sequence. The maximum entropy model overcomes the independence assumption, but suffers from the problem of labeling (the state to be labeled) bias. Conditional random field models were created to solve these problems. A Hidden Markov

Model is a joint probability that (x, y) occurs simultaneously, while a Conditional Random Field is a conditional probability that y occurs when x. Equation (1) gives the Shannon entropy formula and equation (2) gives the properties of entropy.

$$H(X) = -\sum_{x \in X} p(x) \log p(x) \tag{1}$$

$$0 \le H(X) \le \log |X| \tag{2}$$

where X is the number of random variables in a discrete distribution: when X is a deterministic thing, entropy is minimized and the left equation holds; when X is a completely random thing, i.e., uniformly distributed, entropy is maximized and the right equation holds.

The principle of maximum entropy holds that a reasonable probability distribution derived from incomplete information should be the maximum entropy value that satisfies the constraints. Therefore, it is a typical constrained optimization problem. Equation (3) defines the conditional entropy, equation (4) defines the model purpose, and equation (5) defines the characteristic function.

$$H(y|x) = -\sum_{(x,y) \in z} p(y,x) \log p(y|x) \tag{3}$$

$$p^*(y|x) = \arg \max_{p(y|x) \in P} H(y|x) \tag{4}$$

$$f_i(x, y) \in \{0,1\} (i = 1, 2, \cdots, m) \tag{5}$$

Constraints:(1) The total probability is 1, i.e., $\sum_{y=Y} p(y|x) = 1$;

The mathematical expectation is consistent with the empirical expectation, i.e.: $E(f_i) = \tilde{E}(f_i) (i = 1, 2, \cdots, m)$

$$\tilde{E}(f_i) = \sum_{(x,y) \in z} \tilde{p}(x,y) f_i(x,y) = \frac{1}{N} \sum_{(x,y) \in T} f_i(x,y) N = |T| \tag{6}$$

$$\begin{aligned} E(f_i) &= \sum_{(x,y) \in z} p(x,y) f_i(x,y) = \sum_{(x,y) \in z} p(x) p(y|x) f_t(x,y) \\ &= \frac{1}{N} \sum_{x \in T} \sum_{y \in Y} p(y|x) f_t(x,y) \end{aligned} \tag{7}$$

The Lagrange function for this conditionally constrained optimization problem is:

$$\Lambda(p, \vec{\lambda}) = H(y|x) + \sum_{i=1}^{m} \lambda_i \left(E(f_i) - \tilde{E}(f_i) \right) + \lambda_{m+1} \left(\sum_{y=Y} p(y|x) - 1 \right) \tag{8}$$

Max Maki:

$$p_{\lambda}^*(y|x) = \frac{1}{Z_{\lambda}(x)} \exp\left(\sum_{t=1}^{\infty} \lambda_i f_i(x,y) \right) \tag{9}$$

where $Z(x)$ is the normalization factor:

$$Z_{\bar{\lambda}}(x) = \sum_{y \in Y} \exp\left(\sum_{i=1}^{n} \lambda_i f_i(x,y) \right) \tag{10}$$

Probability:

$$p_{\lambda}(y|x) = \frac{1}{Z_{\lambda}(x)} \prod_{i=1}^{n} \exp\left(\lambda_i f_i(x,y) \right) \tag{11}$$

$$P(X_1, X_2, \cdots, X_N) = \frac{1}{Z}\prod_{i=1}^{N}\Psi_i(C_i) \tag{12}$$

where $\Psi_i(C_i)$ is a pairwise function.

A random field is a set of variables that corresponds to the same sample space, the whole of which is called a random field when randomly assigned to each position according to some distribution. There may be some dependence between these variables. A Markov random field corresponds to an undirected graph, where each node on the graph corresponds to a random variable, and the edges between the nodes represent dependencies between the pairs of variables. A Markov random field has the Markov property: distant factors have little influence on the nature of the current factor, i.e., only the influence of the immediate neighbors is considered. If there are still observations for each variable of a given Markov Random Field, the distribution of the Markov Random Field, at that point, is the conditional distribution, and the Markov Random Field at that point is the Conditional Random Field. A conditional random field is a Markov random field given a sequence of observations.

Linear chain CRFs model. Let $x = \{x_1, x_2, \cdots, x_n\}$ denote the observation sequence and $y = \{y_1, y_2, \cdots, y_n\}$ be a finite set of states, according to the basic theory of random fields then we have:

$$p(y \mid x, \lambda) \propto \exp\left(\sum_j \lambda_j t_j(y_{i-1}, y_i, x, i) + \sum_k \mu_k s_k(y_i, x, i)\right) \tag{13}$$

Label the transfer eigenfunctions from position $i-1$ to i: $t_j(y_{i-1}, y_i, x, i)$ and the state eigenfunctions from position i: $s_k(y_i, x, i)$. Unify the two eigenfunctions as $f_j(y_{j-1}, y_i, x, i)$, resulting in Eq. (14) and normalizing the coefficients as Eq. (15).

$$p(y \mid x, \lambda) = \frac{1}{Z(x)}\exp\left(\sum_{i=1}^{n}\sum_j \lambda_j f_j(y_{i-1}, y_i, x, t)\right) \tag{14}$$

$$Z(x) = \sum_j \exp\left(\sum_{i=1}^{n}\sum_j \lambda_j f_j(y_{i-1}, y_i, x, i)\right) \tag{15}$$

The main problem with conditional random fields is that they are troublesome to train and slow to converge. Especially in the case that the current window opening is too large, i.e., the range of near neighbors is too wide. For this reason, when applying the Conditional Random Field model, the word list grammar rules of natural language grammar are added to obtain a simplified CRF-NLG fast model.

3. A Generative Adversarial Network-based Model for English Writing Style Generation and Optimization

3.1. *Theory of techniques for generating adversarial networks*

3.1.1. Generating the overall structure of the adversarial network model. The original generative adversarial network is an unsupervised deep learning model that cleverly combines 2 common machine learning models, one model is mainly responsible for generating data from noise, called the generator; the other model is mainly responsible for determining whether the input data is real data or generated data, called the discriminator. The structure of the generative adversarial network is shown in Figure 2.

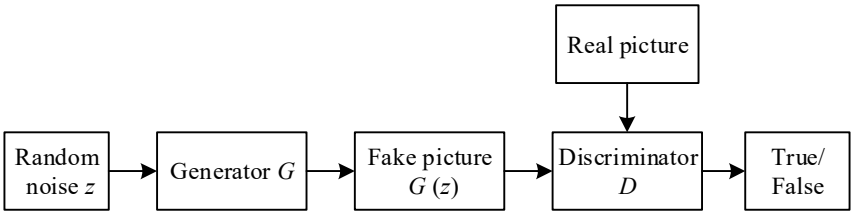


Fig. 2. Generate adversarial network structure

The generator is a network model for generating pictures, its input is a random noise z and then generates pictures from this noise, the generated data is noted as $G(z)$.

The discriminator is a classification network model that categorizes the input images into real and generated images. Its input parameters are x, x represents a picture and the output $D(x)$ represents the probability that x is a real picture, if it is 1, it means that the input picture must be a real picture, while the output is 0, it means that the probability that the input picture is a real picture is zero.

During training, the goal of the generator is to generate images to fool the discriminator, while the goal of the discriminator is to accurately categorize whether the input image is a generated image or a real image. The capabilities of the generator and the discriminator are gradually improved during the training process. In the most ideal case, the generator can produce generated results that are completely indistinguishable by the discriminator, at which point $D(G(z)) = 0.5$.

3.1.2. Principle analysis of generating adversarial network models. The essence of training a generative adversarial network model is that the generator learns the probability distribution of real samples. The goal of this process can be expressed by equation (16).

$$p_g = p_{data} \tag{16}$$

where p_g is the probability distribution of the generated result and p_{data} is the probability distribution of the real sample. In order to achieve this goal, the generative adversarial network constructs a new objective function by introducing a discriminator, and the objective function is shown in equation (17).

$$\begin{aligned} \min_G \max_D V(D,G) = {} & E_{x\sim p_{data}(x)}[\log D(x)] \\ & + E_{z\sim p_z(z)}[\log(1 - D(G(z)))] \end{aligned} \tag{17}$$

where G is the generating function, D is the discriminant function, x is the real data, z is the random noise, and p_z is the probability distribution of the noise. For the discriminator, the two terms on the right side of the equation are the expectation that the discriminator correctly classifies the true sample and the expectation that it correctly classifies the generated result, respectively. So the objective function requires the discriminator to correctly classify the true sample and the generated result. For the generator, only the second term on the right side of the equation is relevant. The second term represents the expectation that the discriminator correctly classifies the generated result, and the generator minimizes this term, essentially making the distribution of the generated result close to the distribution of the true sample.

By alternately optimizing the generator and the discriminator, the generator and the discriminator enhance each other and eventually reach a balance. At this time, the distribution of the generator generated results is close to the distribution of the real sample, and the generator can generate results close to the real sample.

3.2. Model Structure Design

3.2.1. Generators and discriminators：

1) Generator. The generator of this model consists of a combination of UNet network and ViT, and the overall framework structure is shown in Fig. 3.The encoding path of UNet extracts the features on each basic block by reducing the spatial dimensions and enriching the feature dimensions, and then passes these features to the decoding path. The decoding path then transforms these features. Through the preprocessing layer, the image is processed as a tensor of dimension (w_0, h_0, f_0), which the encoding path maps to (w, h, f), where $w = w_0/16$, $h = h_0/16$, $f = 8f_0$. On each basic block in the encoding path, the width and height are halved, and the feature dimensions are doubled on all but the first basic block. Similarly, the decoding path processes the output of the ViT through a series of upsampling and convolutional layers. The dimension and height are doubled on each basic block and the feature dimension is halved. The preprocessing layer consists of a convolution and a leakyReLU activation function, and the postprocessing layer consists of a convolution of 1×1 and a sigmoid activation function.

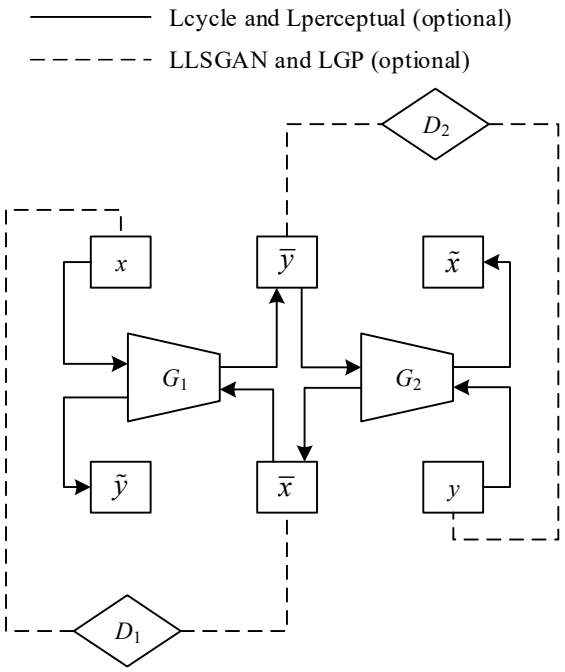


Fig. 3. Schematic diagram of the overall frame structure

ViT connects the end of the encoding path and the beginning of the decoding path of the UNet network.ViT first flattens the spatial dimensions to obtain a feature matrix of dimension $(w\times h, f)$, then connects it to the Fourier positional embedding to obtain a feature matrix of dimension $(w\times h, f+f_p)$, and then goes through the linear layer to map $f+f_p$ to f_v. A Multi Headed Self-Attention Transformer consists of 12 encoder modules. The encoder modules are composed of layer normalization, a multi-head self-attention mechanism, and an FFN. Unlike the original encoder block design, this paper employs rezero regularization by introducing a trainable scale parameter α. Before passing the output to the decoding path, ViT performs a tuning process to recover its spatial dimension. In this paper, the feature dimension $f, f_p, f_v = 384$, linearly extends the dimension $f_h = 4f_v$.

2) Discriminator. For the discriminator, in this paper, we use PatchGAN of 70×70, which is a full convolutional neural network, this discriminator structure has fewer parameters than the full image discriminator, and therefore is faster and more efficient.

3.2.2. Loss function：

1) Fighting against loss. Adversarial loss can make the generated English writing style match the target writing style as much as possible, and can play a role in improving the quality of the writing style generated by the model. For mapping $G_1 : X -> Y$, generator G_1 expects the generated writing style $G_1(x)$ to be as similar as possible to the real data y in domain Y, and discriminator D_2 needs to determine whether the writing style of this input is the source writing style or the generated writing style. In the original CycleGAN model, the loss function for this mapping is denoted as:

$$\begin{aligned} L_{adv}\left(G_1, D_2, X, Y\right) = & E_{y\sim p_{data}(y)}\left[\log\left(D_2(y)\right)\right] \\ & + E_{x\sim p_{data}(x)}\left[\log\left(1-D_2\left(G_1(x)\right)\right)\right] \end{aligned} \tag{18}$$

Generator G_1 goal is to minimize L_{adv}, and discriminator D_2 wants to maximize it, and they play with each other. Similarly, the loss function of mapping $G_2 : Y -> X$ is:

$$\begin{aligned} L_{adv}\left(G_2, D_1, Y, X\right) = & E_{x\sim p_{data}(x)}\left[\log\left(D_1(x)\right)\right] \\ & + E_{y\sim p_{data}(y)}\left[\log\left(1-D_1\left(G_2(y)\right)\right)\right] \end{aligned} \tag{19}$$

However, the adversarial loss is in the form of negative log-likelihood, which makes it difficult for the model to converge during the training process, and thus the negative log-likelihood form is an unreasonable metric. During the experimental process, this paper uses the more stable least squares form to measure the adversarial loss to improve the stability of training and generate high-quality writing styles. Thus.

$$\begin{aligned} L_{LSGAN}\left(G_1, D_2, X, Y\right) = & E_{y\sim p_{data}(y)}\left[\left(D_2(y)-1\right)^2\right] \\ & + E_{x\sim p_{data}(x)}\left[\left(D_2\left(G_1(x)\right)\right)^2\right] \end{aligned} \tag{20}$$

$$\begin{aligned} L_{LSGAN}\left(G_2, D_1, Y, X\right) = & E_{x\sim p_{data}(x)}\left[\left(D_1(x)-1\right)^2\right] \\ & + E_{y\sim p_{data}(y)}\left[\left(D_1\left(G_2(y)\right)\right)^2\right] \end{aligned} \tag{21}$$

2) Cyclic consistency loss. The adversarial loss allows generator G_1 and generator G_2 to learn the distribution of writing style X and writing style Y. But after mapping from the source writing style domain X to the target writing style domain Y, the mapping back to the source writing style domain X must be similar to the original writing style, and the same is true when mapping from writing style domain Y to writing style domain X. The structure of the cyclic consistency loss is shown in Figure 4. The cyclic consistency loss ensures that the cyclic transformation can restore the writing style to its original state, which can be regarded as a regularization, with the strength of the regularization controlled by coefficient λ_{cyc}. After obtaining the forged writing style $G_1(x)$ from the writing style x, the forged writing style is then fed into the generator G_2 to obtain the reconstructed writing style $G_2\left(G_1(x)\right)$, which constrains the reconstructed writing style $G_2\left(G_1(x)\right) = x$, which is the embodiment of the cyclic consistency in this study, and the specific formula is formulated as:

$$\begin{aligned} L_{cycle}\left(G_1, G_2\right) = & E_{x\sim p_{data}(x)}\left[\left\|G_2\left(G_1(x)\right)-x\right\|_1\right] \\ & + E_{y\sim p_{data}(y)}\left[\left\|G_1\left(G_2(y)\right)-y\right\|_1\right] \end{aligned} \tag{22}$$

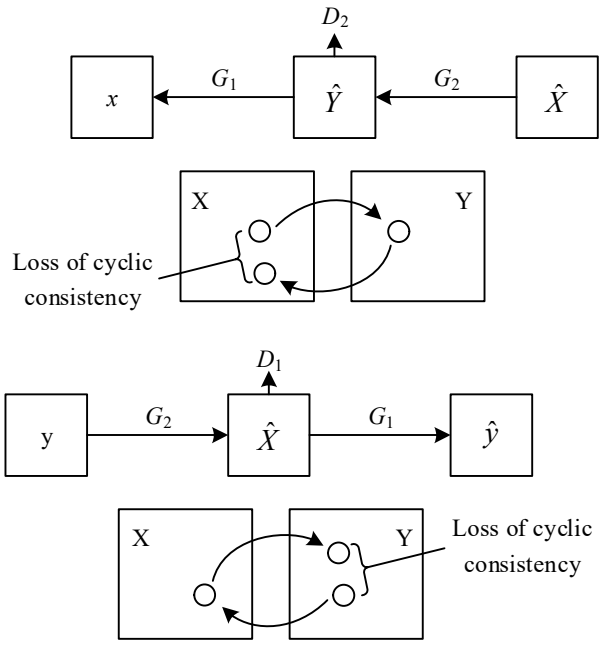


Fig. 4. Cyclic consistency loss

3) Perceptual Loss. Perceptual loss has been widely used as an effective loss in generative tasks. Perceptual loss function is a kind of evaluation loss function, and the perceptual loss function can make a more correct evaluation by comparing the high-level perceptual and semantic differences and output a smaller loss value.

Perceptual loss uses a VGG-16 network pre-trained on the ImageNet dataset to extract writing style features. In this paper, perceptual loss is used as a complement to cyclic consistency loss in order to improve the quality of writing style generation. The perceptual loss is defined as

$$\begin{aligned} L_{perceptual}\left(G_1,G_2\right) &= E_{x\sim p_{data}(x)}\left[\left\|\varphi\left(G_2\left(G_1(x)\right)\right)-\varphi(x)\right\|_1\right] \\ &+E_{y\sim p_{data}(y)}\left[\left\|\varphi\left(G_1\left(G_2(y)\right)\right)-(y)\right\|_1\right] \end{aligned} \tag{23}$$

4) Gradient penalization. The gradient penalty stabilizes the training process and improves the quality of the generated writing style, the gradient penalty term is shown in the following equation:

$$\begin{aligned} L_{GP} &= E_{\hat{\lambda}\sim p_{data}(\hat{x})}\left[\left(\left\|\nabla_{\hat{x}}D_1(\hat{x})\right\|_2-\gamma\right)^2/\gamma^2\right] \\ &+E_{\hat{\gamma}\sim p_{data}(\hat{y})}\left[\left(\left\|\nabla_{\hat{y}}D_2(\hat{y})\right\|_2-\gamma\right)^2/\gamma^2\right] \end{aligned} \tag{24}$$

where $p_{data}(\hat{x})$ and $p_{data}(\hat{y})$ are distributions uniformly sampled along a straight line between pairs of points sampled from the real and generated data distributions.

5) Total loss. The total loss function of the model is a linear combination of each of the above loss functions, with the weights of each loss function adjusted by hyperparameters:

$$\begin{aligned} L_{total} &= L_{LSGAN}\left(G_1,D_2,X,Y\right)+L_{LSGAN}\left(G_2,D_1,Y,X\right) \\ &+\lambda_{cyc}L_{cycle}\left(G_1,G_2\right)+\lambda_{perceptual}L_{perceptual}\left(G_1,G_2\right)+\lambda_{GP}L_{GP} \end{aligned} \tag{25}$$

4. Performance of English writing style generation and optimization models

4.1. Writing style identification

In order to further validate the effectiveness of the research method of this paper in identifying writing styles, eight classic English and American literary works were chosen to be explored: Pride and Prejudice, Hamlet, Moby Dick, 1984, Wuthering Heights, King Lear, Jane Eyre, and Madame Bovary, and the eight works represent different literary styles and language features.

This study focuses on eight experiments, in which any seven of the eight works are used as the training set and the remaining one as the test set. In order to comprehensively evaluate the performance of the model, three metrics, accuracy (P), recall (R) and F1 value (F), are used in this paper. The accuracy rate is the proportion of texts predicted by the model as a certain style that actually belong to that style, the recall rate is the proportion of all texts actually belonging to a certain style that are correctly predicted as that style, and the F1 value is the reconciled average of the accuracy rate and the recall rate for a comprehensive measure of the model's performance.

The style recognition results are shown in Table 1. As can be seen from Table 1, the F1 value for recognizing author's style using the method proposed in this paper is as high as 98.95, and the average F1 value reaches 92.47, which proves that writing style recognition using the method proposed in this paper is reliable.

Table 1. Results of writing style identification(%)

Experiment number	P	R	F
1	80.16	95.39	87.91
2	87.33	92.15	90.24
3	93.01	98.38	96.32
4	98.32	97.29	98.04
5	99.20	98.27	98.95
6	93.14	88.38	91.03
7	82.98	89.29	85.78
8	93.19	89.05	91.47

4.2. Writing style generation

4.2.1. Data sets. This paper collects a collection of works by five authors, Shakespeare, Jane Austen, Mark Twain, Emily Dickinson and Charles Dickens. Samples containing meaningless content such as garbled codes and special characters are directly deleted when performing data cleaning in order to ensure the usability of the data, the intuition behind this practice is that content such as garbled codes are likely to be intentionally textual in the original works, and that the appearance of meaningless content due to coding errors during the process of document e-textualization and crawling of the data, and that deletion of these insiders alone may reduce the usability of the data, therefore, this paper directly removes such samples. The processed data is used as the training set for this experiment.

In this paper, we use the comment text of ASAP evaluation dataset (hereinafter referred to as ASAP dataset) as the test set, the ASAP dataset contains articles in different fields, such as science and technology, culture, art, etc., to ensure that a variety of styles are covered, and 5000 samples are selected from the dataset for the experiments in this paper.

4.2.2. Evaluation indicators：

1) Style Strength. In mood and formal and informal style generation, the style classifier is usually used for the assessment of style strength because positive and negative mood and formal and informal are an explicit, single attribute, while writing style is an implicit, compound, and difficult-to-define

attribute that is deeply embedded in the text. Therefore, in order to measure writing style, this paper improves the classifier's categorization algorithm. Firstly, a writer attribution classifier cls(x) is trained, for all the collection corpus of different writers are given a label corresponding to the writer respectively, the collection is input into the classifier as a feature to get the prediction result, and the loss is calculated with the corresponding label and the prediction result followed by gradient descent to update the classifier parameters.

The trained classifier is able to determine to which writer a text belongs, however, such a classifier cannot be directly used to measure style, because the training data features the entire text, not the style attributes, but also contains a variety of attributes such as genre, subject characters, etc. The unstylized generated text is fed into the classifier, at which point the highest dimension of the n-dimensional probability of taking the classifier's output is unintentional, since the text does not actually belong to any writer, but the relative rate of change can be calculated based on this probability. Specifically, the original text and the sentence after style generation are input into the classifier to get n+1 n-dimensional probability vectors, for each probability vector in the n-dimensional probability vector calculate the quotient of each dimension of probability, and take the dimension with the highest quotient as the classification result of the classifier. Each dimension of the output probability vector of the classifier represents the probability that the text belongs to a certain writer, if the text generated by the style is more in the style of a certain writer, then the probability of the corresponding dimension should rise, so take the dimension with the highest rate of increase as the classification result, i.e., it quantifies the writing style.

2) Content Preservation. Considering that text style generation draws on the idea of sequence-to-sequence tasks such as machine translation, this paper uses BLEU, a commonly used evaluation metric for column-to-sequence tasks, which was first used to evaluate the quality of translations between different languages, but is essentially a calculation of the similarity of sentences in the same language, and can therefore be used for the evaluation of text style generation tasks.

3) Degree of fluency. The fluency of the generated text is evaluated using the Perplexity Level (PPL).

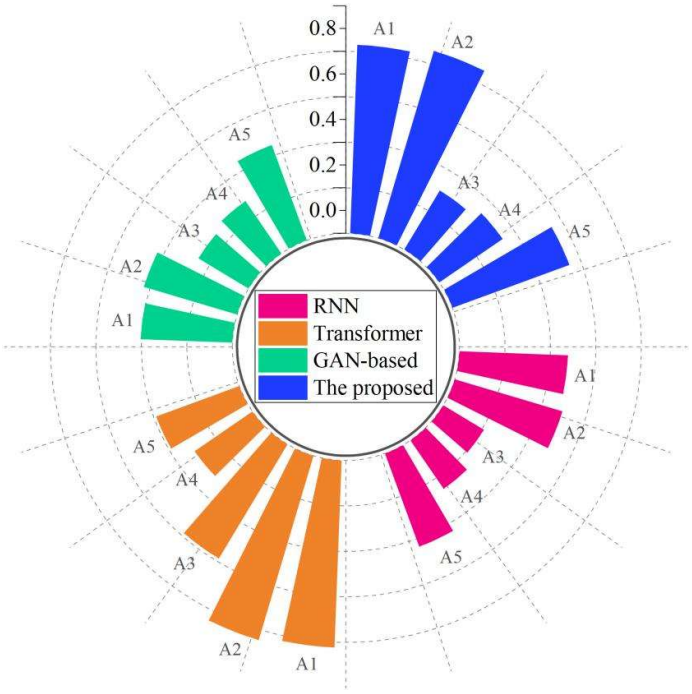
4.2.3. Comparison experiments. In this paper, the proposed model is compared and experimented with three baseline models, which are the traditional RNN-based style generation model, the Transformer-based style generation model, and the existing GAN-based style generation model. The results of the experiments on the ASAP comment dataset are shown in Table 2, where higher scores for style strength and content preservation represent stronger performance, and the opposite is true for fluency.

Table 2. Performance comparison of different models

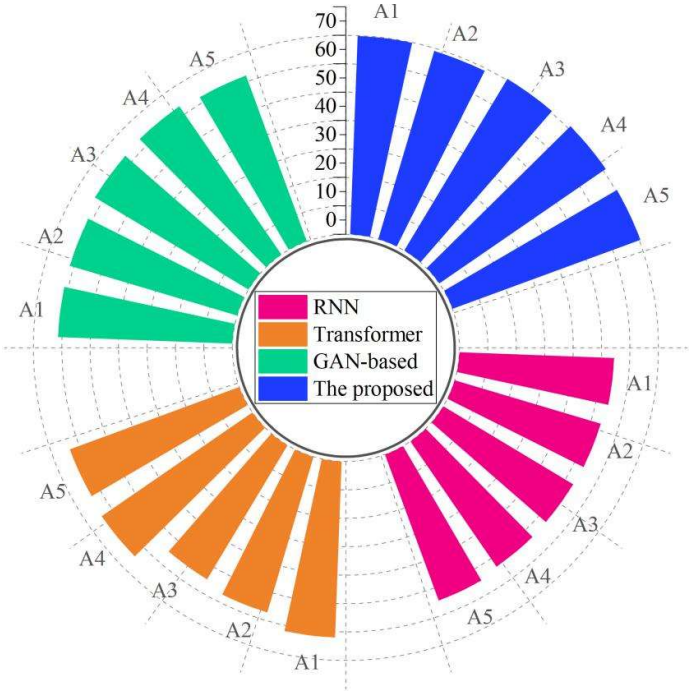
Name of model	Strength of style(↑)	Content preservation(↑)	Smoothness(↓)
RNN	0.3752	53.92	147.2
Transformer	0.5907	60.39	103.7
GAN-based	0.3628	62.24	72.1
The proposed	0.5753	70.15	43.9

Plotting the data in Table 2 with 5-bit writers numbered A1~A5, the resultant details are visualized as shown in Fig. 5, and Fig. 5(a)(b)(c) shows the accuracy comparison data in the dimensions of style strength, content preservation, and fluency, respectively. From Fig. 5(b)(c), it can be seen that the proposed model in this paper shows the strongest performance in the two dimensions of content preservation and fluency, with an improvement of 12.71% and 39.11% on the ASAP dataset, respectively, compared to the existing GAN-based style generation model. As can be seen from Fig. 5(a), the text generated by the Transformer-based style generation model exhibits the highest style strength, but its prediction accuracy does not match the trend shown by the other three models, indicating that the models are confused during the training process, mistakenly believing that the entire training set

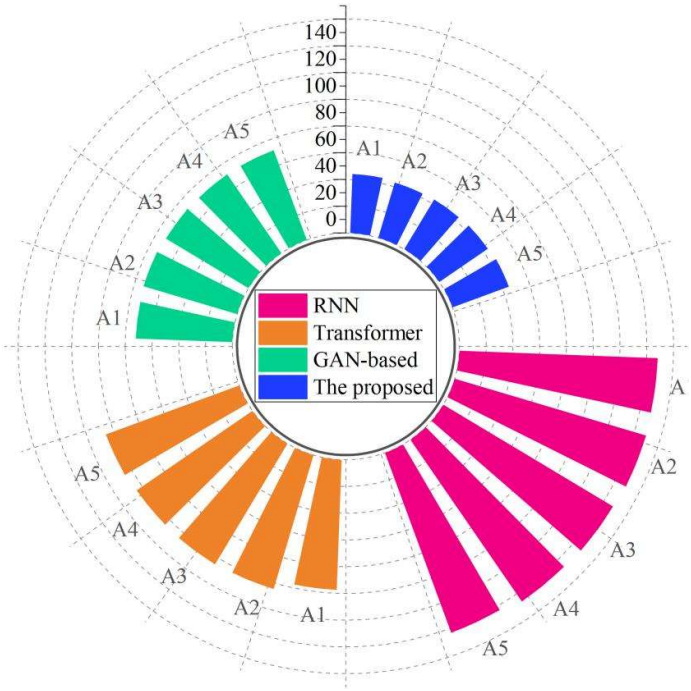
represents the same writing style. The model proposed in this paper, on the other hand, matches the predictive trend of the other two models and has the best overall performance.



(a)Strength of style



(b)Content preservation



(c)Smoothness

Fig. 5. ASAP data set results detail visualization

In this paper, we also further investigate the effect of adopting CRF-NLG model on the results by experimenting the original CRF model and the CRF-NLG model embedded with natural language grammar on the ASAP dataset respectively, and the experimental results are shown in Table 3. From the table, it can be seen that all the metrics show better performance after adopting the CRF-NLG model, which indicates that adopting the CRF-NLG model for the recognition and extraction of professional English terms is helpful for the generation of English writing styles.

Table 3. Effects of CRF-NLG model on experimental results

Model	Strength of style(↑)	Content preservation(↑)	Smoothness(↓)
CRF	0.4786	65.39	59.2
CRF-NLG	0.5753	70.15	43.9

4.3. Application effects

4.3.1. Experimental design. In this paper, a laboratory field user experiment was conducted, and the purpose of this experiment was to assess the effectiveness of an English writing style generation model for assisting real writing tasks. Specifically, this experiment aimed to examine (1) whether the English writing style generation model based on Generative Adversarial Networks is consistent with users' linguistic perceptions and intuitively easy to use, and (2) whether the quality of the users' expressions is optimized after the model has been utilized for writing style generation.

Five authors with different English writing skills participated in the experiment. They were all graduate students majoring in English language and literature, and used the English writing style generation model proposed in this paper to write or revise their papers in practice and mark up their documents. At the end of the experiment, each modification was evaluated by native English speaking experts, and was labeled as “better”, “worse”, or “both” according to the change in the quality of expression before and after the modification (i.e. both expressions before and after the modification were acceptable). The relevant materials in the field of English language and literature were used as

the experimental corpus, and the specific statistical information is shown in Table 4.

Table 4. Statistics of corpus Information

Subdomain name	Total number of meetings/journals	Paper total	Sentence total
English Literature	32	28,938	6,246,537
American Literature	37	31,984	7,974,286
Comparative Literature	23	29,324	2,864,972
Literary Theory	45	40,132	13,762,873
Poetics	19	8,983	1,428,435
Feminist Literature	31	28,398	5,386,242
Romantic Literature	27	24,324	4,286,008
Literary Criticism	43	50,498	12,472,324
Postcolonial Literature	22	19,324	3,972,905
Cultural Studies	24	21,529	2,983,028
Total	303	283,434	61,377,610

4.3.2. Experimental results. The results of the experiment are presented visually in Figure 6. 5 authors made a total of 119 revisions with the help of the English writing style generation model. Ninety-two (77.3%) were evaluated by native English speaking experts as better, 17 (14.3%) as average, and only 11 (9.2%) as worse. The above experimental results preliminarily indicate that more than 90% of the revisions produced by the authors with the help of the system are acceptable, and that the English writing style generation model is helpful in improving the quality of writing.

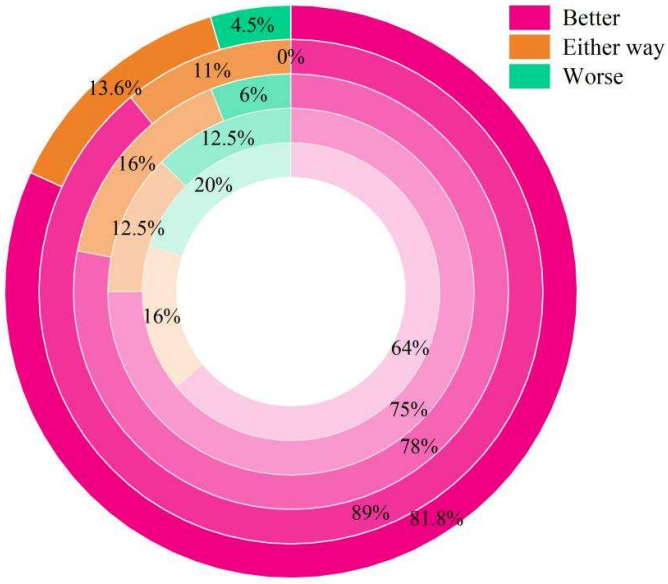
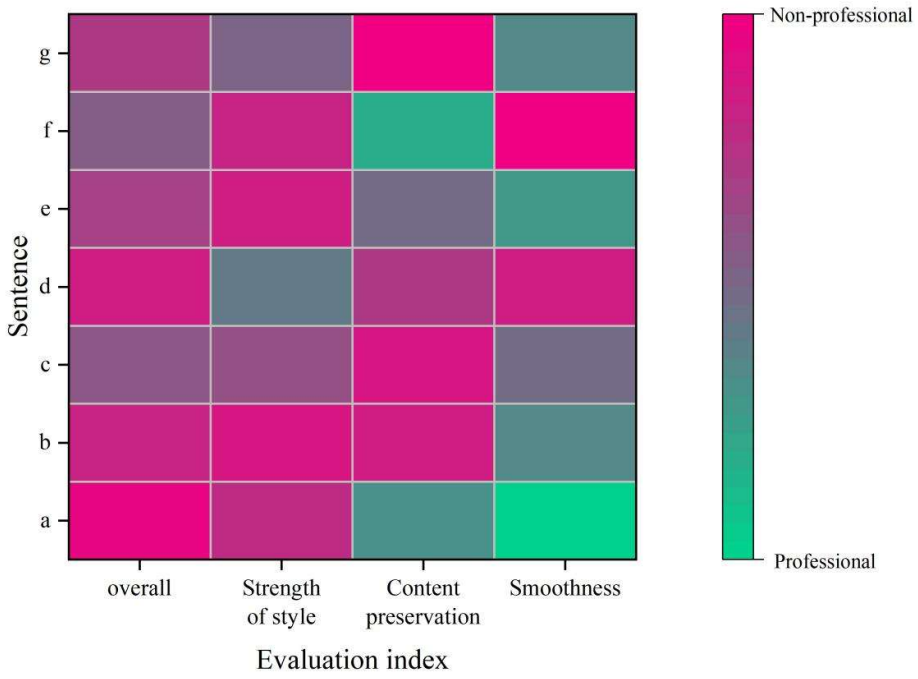


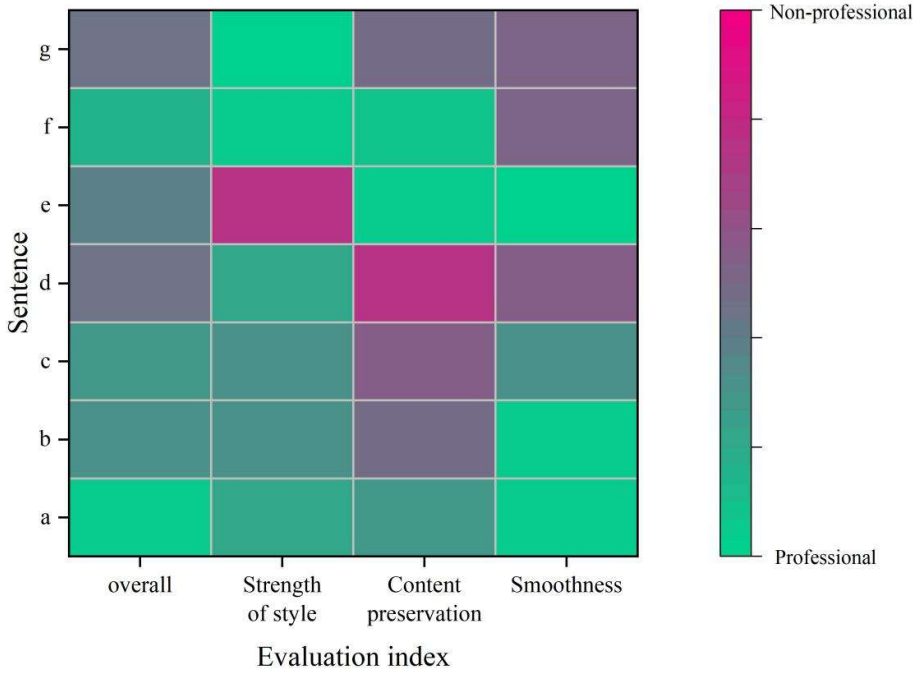
Fig. 6. Modification quality distribution of users with different writing experience

The above results also suggest that the few cases where users use the model may also produce poorer quality modifications. The overall and detailed style scores before and after the 119 revisions are shown in Figure 7. Further analysis revealed that #1 and #2, who lacked writing experience, generated a total of 8 incorrect usages, while users #3, #4, and #5, who were relatively experienced in writing, generated a total of 3 incorrect usages. User feedback indicated that Nos. 1 and 2 had difficulty

evaluating the quality of the generated results due to their lack of English proficiency. However, the other 3 relatively experienced subjects in English were more adept at relying on their own linguistic intelligence to make further judgments. From this, it can be inferred that the difference in the writers' own language proficiency and writing experience would lead to different interaction patterns with the writing assistance methods. In the future, the writing assistance method needs to be further optimized for users with different English proficiency, so as to achieve a more personalized writing style optimization.



(a)Before



(b)After

Fig. 7. Visual comparison of writing style before and after modification(part)

We also asked five subjects about their subjective feelings about using the model. For the English writing style generation model based on Generative Adversarial Networks, users usually only need a 5-minute trial to familiarize themselves with its usage. Users indicated that the English writing style generation model proposed in this paper is intuitive and easy to understand. User feedback suggests that the proposed model is easy to use, generates results with high auxiliary value, and can help writers greatly improve their writing confidence.

5. Conclusion

This paper proposes an English writing style generation model based on generative adversarial networks, and evaluates its performance in three dimensions: writing style recognition, writing style generation and application effect.

The F1 value for recognizing author's style using the model proposed in this paper reaches as high as 98.95, and the average F1 value reaches 92.47. The model proposed in this paper shows the strongest performance in the dimensions of content preservation and fluency, which are improved by 12.71% and 39.11%, respectively, compared with the existing GAN-based style generation model. All the metrics show better performance after the introduction of the CRF-NLG model, which suggests that the use of the CRF-NLG model for the identification and extraction of professional English terms is beneficial to the generation of English writing styles.

In the application, 92 (77.3%) of the 119 modifications were better, 17 (14.3%) were both, and only 11 (9.2%) were worse, and users usually only need 5 minutes of trial to familiarize themselves with the way of using this model. This proves that the model proposed in this paper is easy to use and generates results of high auxiliary value.

References

[1] Salvagno, M., Taccone, F. S., & Gerli, A. G. (2023). Can artificial intelligence help for scientific writing?. Critical care, 27(1), 75.

[2] Fitria, T. N. (2023, March). Artificial intelligence (AI) technology in OpenAI ChatGPT application: A review of ChatGPT in writing English essay. In ELT Forum: Journal of English Language Teaching (Vol. 12, No. 1, pp. 44-58).

[3] Chen, T. J. (2023). ChatGPT and other artificial intelligence applications speed up scientific writing. Journal of the Chinese Medical Association, 86(4), 351-353.

[4] Dergaa, I., Chamari, K., Zmijewski, P., & Saad, H. B. (2023). From human writing to artificial intelligence generated text: examining the prospects and potential threats of ChatGPT in academic writing. Biology of sport, 40(2), 615-622.

[5] Khan, A. L., Hasan, M. M., Islam, M. N., & Udin, M. S. (2024). Artificial intelligence tools in developing English writing skills: Bangladeshi university EFL students' perceptions. English Education: Jurnal Tadris Bahasa Inggris, 17(2), 345-371.

[6] Zhao, X. (2023). Leveraging artificial intelligence (AI) technology for English writing: Introducing wordtune as a digital writing assistant for EFL writers. RELC Journal, 54(3), 890-894.

[7] Giglio, A. D., & Costa, M. U. P. D. (2023). The use of artificial intelligence to improve the scientific writing of non-native English speakers. Revista da Associação Médica Brasileira, 69(9), e20230560.

[8] Lu, X. (2019). An empirical study on the artificial intelligence writing evaluation system in China CET. Big data, 7(2), 121-129.

[9] Charpentier-Jiménez, W. (2024). Assessing artificial intelligence and professors' calibration in English as a foreign language writing courses at a Costa Rican public university. Actualidades Investigativas en Educación, 24(1), 533-557.

[10] Al Mahmud, F. (2023). Investigating EFL students' writing skills through artificial intelligence: Wordtune application as a tool. Journal of Language Teaching and Research, 14(5), 1395-1404.

[11] Nazari, N., Shabbir, M. S., & Setiawan, R. (2021). Application of Artificial Intelligence powered digital writing assistant in higher education: randomized controlled trial. Heliyon, 7(5).

[12] Creswell, A., White, T., Dumoulin, V., Arulkumaran, K., Sengupta, B., & Bharath, A. A. (2018). Generative adversarial networks: An overview. IEEE signal processing magazine, 35(1), 53-65.

[13] Wang, Z., She, Q., & Ward, T. E. (2021). Generative adversarial networks in computer vision: A survey and taxonomy. ACM Computing Surveys (CSUR), 54(2), 1-38.

[14] Lee, M., & Seok, J. (2019). Controllable generative adversarial network. Ieee Access, 7, 28158-28169.

[15] Aldausari, N., Sowmya, A., Marcus, N., & Mohammadi, G. (2022). Video generative adversarial networks: a review. ACM Computing Surveys (CSUR), 55(2), 1-25.

[16] Sajeeda, A., & Hossain, B. M. (2022). Exploring generative adversarial networks and adversarial training. International Journal of Cognitive Computing in Engineering, 3, 78-89.

[17] Jabbar, A., Li, X., & Omar, B. (2021). A survey on generative adversarial networks: Variants, applications, and training. ACM Computing Surveys (CSUR), 54(8), 1-49.

[18] Chavdarova, T., & Fleuret, F. (2018). Sgan: An alternative training of generative adversarial networks. In Proceedings of the IEEE conference on computer vision and pattern recognition (pp. 9407-9415).

[19] Karras, T., Aittala, M., Hellsten, J., Laine, S., Lehtinen, J., & Aila, T. (2020). Training generative adversarial networks with limited data. Advances in neural information processing systems, 33, 12104-12114.

[20] Gurumurthy, S., Kiran Sarvadevabhatla, R., & Venkatesh Babu, R. (2017). Deligan: Generative adversarial networks for diverse and limited data. In Proceedings of the IEEE conference on computer vision and pattern recognition (pp. 166-174).

[21] Zhou, B., Zhou, Q., & Li, Z. (2024). Addressing data imbalance in crash data: evaluating Generative Adversarial Network's efficacy against conventional methods. IEEE Access.

[22] Zhang, K. (2021). On mode collapse in generative adversarial networks. In Artificial Neural Networks and Machine Learning–ICANN 2021: 30th International Conference on Artificial Neural Networks, Bratislava, Slovakia, September 14–17, 2021, Proceedings, Part II 30 (pp. 563-574). Springer International Publishing.

[23] Lala, S., Shady, M., Belyaeva, A., & Liu, M. (2018). Evaluation of mode collapse in generative adversarial networks. High performance extreme computing.

[24] Liu, Q. (2023). Text complexity analysis of Chinese and foreign academic English writing via mobile devices based on neural network and deep learning. Library Hi Tech, 41(5), 1317-1332.

[25] Wang, P., & Huang, X. (2022). An online English writing evaluation system using deep learning algorithm. Mobile Information Systems, 2022(1), 7605989.

[26] Chen, B., Bao, L., Zhang, R., Zhang, J., Liu, F., Wang, S., & Li, M. (2024). A multi-strategy computer-assisted EFL writing learning system with deep learning incorporated and its effects on learning: A writing feedback perspective. Journal of Educational Computing Research, 61(8), 60-102.

[27] Cheng, L., Ben, P., & Qiao, Y. (2022). Research on automatic error correction method in English writing based on deep neural network. Computational Intelligence and Neuroscience, 2022(1), 2709255.

[28] Qin, F. (2022). College english intelligent writing score system based on big data analysis and deep learning algorithm. Journal of Database Management (JDM), 33(5), 1-26.

[29] Cui, Y., & Li, H. (2022). Evaluation system of mobile english learning platform by using deep learning algorithm. Mobile Information Systems, 2022(1), 3849079.

[30] Guo, J. (2022, July). A Study of College English Writing from the Perspective of Deep Learning. In EAI International Conference, BigIoT-EDU (pp. 1-9). Cham: Springer Nature Switzerland.

[31] Cho, Y., & Kim, J. (2021). Production of mobile english language teaching application based on text interface using deep learning. Electronics, 10(15), 1809.

[32] Li, Y. (2022). Deep Learning-Based Correlation Analysis between the Evaluation Score of English Teaching Quality and the Knowledge Points. Computational Intelligence and Neuroscience, 2022(1), 4102959.

[33] Jing, Y. (2023, November). Design of English Assisted Learning System Based on Deep Learning and Data Analysis Techniques. In 2023 International Conference on Computer Simulation and Modeling, Information Security (CSMIS) (pp. 315-319). IEEE.