Scalable Computing: Practice and Experience, ISSN 1895-1767, http://www.scpe.org
© 2024 SCPE. Volume 25, Issues 3, pp. 2088–2102, DOI 10.12694/scpe.v25i3.2770

DESIGN AND APPLICATION OF AUTOMATIC ENGLISH TRANSLATION GRAMMAR ERROR DETECTION SYSTEM BASED ON BERT MACHINE VISION[*]

YU QING[†]

Abstract. Given the traditional handwritten English fonts, the accuracy of grammar error detection is unsatisfactory. This attribute leads to poor grammar error correction. Based on the optimized BERT machine vision model, an automatic English translation grammar error detection system is proposed in this paper. First, the basic architecture of the Transformer model and BERT model is considered, and a mixed attention module is discussed into the Transformer model to capture the features of the target in space and channel dimensions and realize the modeling of the context dependence of the target features. The feature maps are sampled by multiple parallel only if then cavity convolutions with different void rates to obtain multi-scale features and enhance local feature representation. Then, the input words of the BERT model are weighted by TFIDF to improve the feature extraction ability of the BERT model and construct the TF-BERT model. A database query rewriting model based on BERT and Transformer is proposed. The construction details of the model are described from the aspects of encoding processing, table embedding, and decoding processing respectively. Based on the principles of English translation, we extract grammatical features and build a grammar error detection method. TF-BERT model is selected as the basic framework. Combined with the hybrid attention mechanism, an automatic error correction model of English grammar is constructed. Finally, it is found that the loss value of the traditional system is as high as 0.7411, and the accuracy rate is 75%, while the loss value of the English grammar error detection system proposed in this paper is 0.2639, and the accuracy rate is 100%, which is 25% higher than that of the traditional system, and the performance is remarkable.

Key words: BERT; Mixed attention; Transformer model; Empty convolution; Grammar error correction;

1. Introduction.

The quality of English translation is mainly determined by the total number of incorrect texts such as grammatical errors and spelling errors. English translation usually pays more attention to the improvement of translation efficiency, the lightweight upgrade of products and the better user experience, and pays less attention to the quality control of translation [1]. As a result, there are some errors in the translation results. In this case, the reviewer generally needs to implement manual proofreading, which is not only time-consuming and laborious, but also often unable to eliminate all incorrect texts [2]. Therefore, an English translation robotic error detection machine is studied to realize automated proofreading of English translation and enhance the effectiveness and satisfaction of English translation.

At present, Liu Yakui et al. proposed a mechanical function extraction approach primarily based on laptop vision. First, a high-speed image acquisition system is used to collect the motion trajectory of the mechanism, and then Hough transform is used to automatically locate the key corner points in the video image [3]. Wu Yiquan et al. comprehensively reviewed PCB (Printed Circuit Board) defect detection algorithms primarily based on laptop imagination and prescient in the previous 10 years from the three dimensions of usual photo processing, normal computer gaining knowledge of and deep learning, and analyzed their blessings and negative aspects [4]. Zhou Qihong et al. used industrial cameras to capture images of yarn being sucked into the nozzle, improved gray enhancement methods to increase yarn features and background contrast, used the Canny operator for edge detection, and finally obtained yarn image features by dividing the upper and lower regions of interest and optimizing Hof line detection, and extracted the required position information by positioning algorithm [5]. Li Yong et al. solved the problem that the conventional measurement algorithm could not determine the measurement points corresponding to the irregular deformation contour and the region of the higher-order transition curve could not be measured, and realized the intensive measurement width in the axis direction to approximate the true edge width distribution to the greatest extent [6]. Ruan Jie et al.

[*]

[†] The school of general education, ChongQing Industry Polytechnic College, Chongqing 401120, China (yuqingedu@outlook.com)

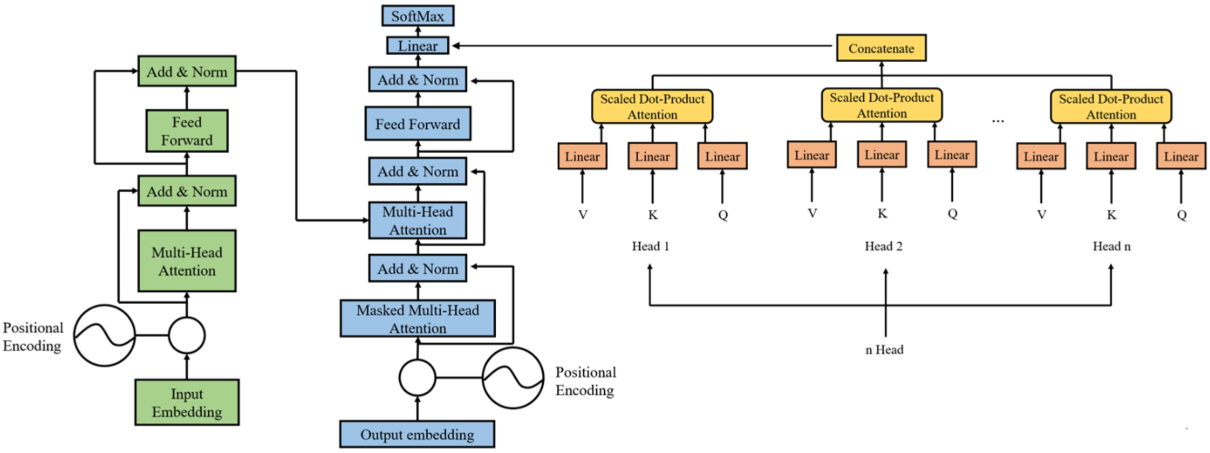


Fig. 2.1: Transformer network structure

converted the information of the template image from the image domain to the frequency domain based on a two-dimensional discrete cosine transform, selected the coefficient matrix represented by a low-frequency signal, converted the matrix into hash value through a hash algorithm, and adopted template matching method for each input frame image to track measurement points and extract pixel coordinates of measurement points [7].

Based on the optimized BERT machine vision model, an automatic English translation grammar error detection system is proposed in this paper. The basic architecture of the Transformer model and BERT (Bidirectional Encoder Representations from Transformers) model is explained, and a mixed attention module is introduced into the Transformer model to capture the features of the target in space and channel dimensions and realize the modeling of the context dependence of the target features. Then, the feature maps are sampled by multiple parallel cavity convolutions with different void rates to obtain multi-scale features and enhance local feature representation. Then, the input words of the BERT model are weighted by TFIDF to improve the feature extraction ability of the BERT model and construct the TF-BERT (Transformer Bidirectional Encoder Representations from Transformers) model. A database query rewriting model based on BERT and Transformer is proposed. The construction details of the model are described from the aspects of encoding processing, table embedding, and decoding processing respectively. Based on the principles of English translation, we extract grammatical features and build a grammar error detection method. The TF-BERT model is selected as the basic framework and combined with the mixed attention mechanism; the automatic error correction model of English grammar is constructed.

2. Optimized BERT model.

2.1. Transformer model. The combination of CNN (Convolutional Neural Network) and LSTM (Long Short-Term Memory) network and the Seq2Seq model has shown good results in many tasks in the field of natural language processing. The use of CNN can effectively extract local information, and the use of the Seq2Seq model based on the LSTM network can well process serialized data [8]. However, limited by the network architecture of LSTM itself, it is still difficult to deal with the problem of long-distance dependence, and the model cannot be parallelized, and the training time is long. To overcome these shortcomings of LSTM, Vaswani et al. 3 proposed a Transformer network model in 2017, which only adopts a self-attention mechanism to build, which not only reduces network parameters and computation amount but also improves parallel computing efficiency. Additionally, it performed better on the majority of tests in the area of natural language processing [9].

The Transformer network design is separated into encoder and decoder portions, just like the Seq2Seg architecture in Figure 2.1.

Among them, an Encoder or Decoder is composed of multiple encoder modules or multiple decoder modules independently stacked together layer by layer. The Encoder part is mainly responsible for extracting the relationship inside the input sequence and modeling the sequence, which can be regarded as the sequence feature extractor. The Decoder section can not only extract features but also build language models and sequence generation models. When the Transformer model receives input vectors. Firstly, the input vector is positionally encoded, relative position information and absolute position information are added, and then the position-encoded vector is input into the Encoder. After one layer of multi-attention layer, sub-residual connection and layer normalization are performed, and then enter into the subsequent layer of thoroughly related feedforward network, and then residual connection and layer normalization are carried out again, the output result of Encoder is obtained, and the result is output to Decoder [10]. In general, every small Encoder layer consists of a self-attentional computing layer and a completely related feedforward community layer for function mapping. The entry of the Decoder phase is additionally a vector encoded by using position, and then it goes via two consecutive multi-sensing layers, residual connection, and layer normalization operations, and then it is entered into the linked feedforward network. Finally, the last output is received through the processing of the linear layer. Decoder is comparable to Encoder; however, Decoder provides a masks multi-perception layer between the self-attention layer and the linked feedforward community layer, that is, it introduces a masks matrix with the values of the top triangle being terrible infinity and the values of the decrease triangle being 0, which serves to masks the future phrases in the sequence. To assist the mannequin, operate higher in the prediction phase.

In essence, the Transformer model is still a stack of multiple network layers, and the main reason why the Transformer model is so outstanding is the introduction of the following four key technologies.

1. Location coding
   LSTM, through its complex gating mechanism and cell state, is able to efficiently process position information in the sequence and transfer important information between different parts of the sequence. However, because Transformer uses a completely different self-attention mechanism to build its model, it has no location information itself. Therefore, it is necessary to introduce position coding to preprocess the input sequence and obtain the relative position and absolute position information of each element in the sequence [11]. The commonly used calculation method of position relation is as follows.

$$PE_{(pos,2i)} = \sin\left(pos/10000^{2i/d_{model}}\right) \tag{2.1}$$

$$PE_{(pos,2i+1)} = \cos\left(\mathrm{pos}/10000^{2i/d_{model}}\right) \tag{2.2}$$

2. Multi-head perception
   The multi-head perception structure maps Q, K and V in Attention linearly many times, and then scales the mapped matrix to get the value of self-attention. If the above operation is repeated n times, the calculation results of n heads can be obtained, and these calculations can be executed in parallel, where the Attention used is the operation of self-attention, and the calculation method of Multi-head attention is as follows:

$$\text{MultiHead\_Attention}(Q, K, V) = \text{Concat}\left(\text{head}_1, \ldots, \text{head}_i\right) W^o \tag{2.3}$$

3. Residual connection and normalization
   To avoid the problem of gradient disappearing due to the deepening of network layers during the training process, the Transformer model draws on the idea of a residual network and adds a residual module between every two adjacent Encoder layers. That is, the input part x and the output part of the multi-head sensing structure are added directly, and then a layer normalization processing module is connected. The whole process can be expressed in the following formula.

$$LayerOutput = LayerNorm(x + MultiHead\_Attention(x)) \tag{2.4}$$

Table 2.1: Model parameters table

model	N	dmodel	h	Total parameter
BERTbase	12	768	12	1.1
BERTlarge	24	1024	16	3.4

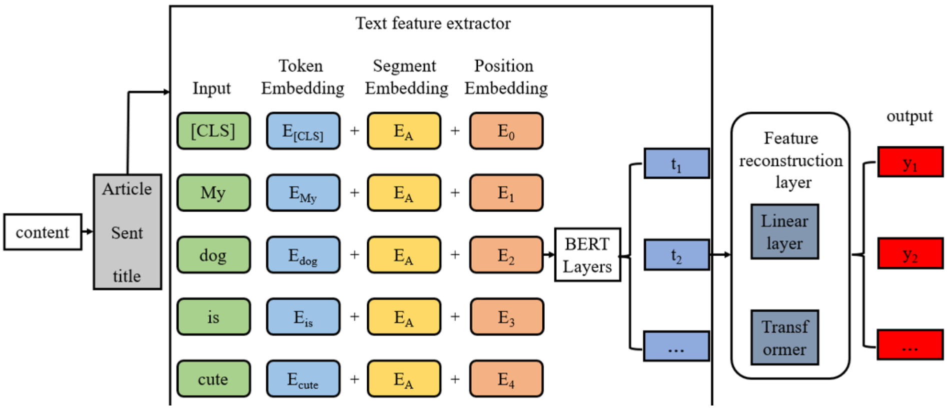


Fig. 2.2: Input structure of the BERT model

4. Position oriented fully connected feedforward network
   To spatially map the outcomes of various senses and improve the model's capacity to extract features, each layer of encoder and decoder, which incorporates a fully linked feedforward network, is applied to each place. The network is made up of a ReLU activation function and two linear transformations as shown below.

$$FFN(x) = max\left(0, W_1x + b_1\right)W_2 + b_2 \tag{2.5}$$

2.2. BERT model. The basic framework of the BERT language model is essentially to construct a multi-layer bidirectional Encoder network using the Transformer coding part. In other words, the BERT model uses a dual Transformer to learn. Google released two versions of BERT model BERT$_{base}$ and BERT$_{large}$, and their feedforward size is set to four layers [12].

BERT$_{base}$ has 12 coding modules, i.e. N=12, each coding module has 12 headers from the attention operator module, h=12. The entry vector dimension is 768 which is d$_{model}$= 768. In BERT$_{large}$, N=24, h=16. d$_{model}$=1024. These differences bring the total number of parameters for BERT$_{base}$ to 110 million and BERT$_{large}$ to 340 million, as shown in Table 2.1.

There are also some differences between the BERT model and the Transformer model in the processing of input sequences. The BERT model can represent a single sentence or a pair of sentences in an input sequence when facing different tasks. For each Token, its input can be generated by adding segment embedding and positional embedding with text corresponding to each other. As shown in Figure 2.2 below.

To separate two sentences, indicating the end of the previous sentence and the beginning of the last sentence.

In addition to word embedding in the input sequence, BERT also adds position encoding like Transformer but does not use Transformer's sine and cosine encoding. Instead, the location information of each Token in the corpus is randomly initialized with the learned location embedding and then updated gradually with the training of the model. BERT model also adds clause embedding, each sentence uses a sentence entry, mainly to learn the relationship between multiple sentences and text matching pre-training, such as sentence relationship inference.

The purpose of pre-training is to train a model that can handle various downstream tasks as much as possible, and the parameters that need to be adjusted when using this model for transfer learning should be as few as possible [13]. Generally speaking, pre-training is to accumulate experience in various occasions and environments, so that the model can remember each word in a certain context and semantic usage so that the model can handle new tasks according to experience when carrying out transfer learning.

The fine-tuning of BERT includes sequential arbitrary and block tasks, in which there are subpairs and single categories in the column, the sentence beginning of which uses [CLSI], and only needs to take out the encoded vector for the downstream task [14]. The block task also includes the question answering task and sequence labeling task, which do not need to consider the output of [CLSI], but only need the vectorization result of word coding. The question-answering task can judge the position of the answer according to the coding vector, and the sequence-labeling task can predict the word labeling according to the coding vector.

2.3. Mixed attention design. Using self-attention, the location attention module may encode more general context information into local characteristics, improving its capacity to represent spatial information.

The spatial connection between any two feature pixels is modeled by the matrix.

$$T_{Sji} = \frac{\exp\left(T_{Bi} \cdot T_{Cj}\right)}{\sum_{i=1}^{N} \exp\left(T_{Bi} \cdot T_{Cj}\right)} \tag{2.6}$$

At the same time, input feature T_A is input into the convolution layer to generate a new feature map

The convolution layer receives input feature TA at the same time, creating a new feature map $\mathrm{T}_D \in \mathrm{R}^{C\times H\times W}$ and reassembling it into $\mathrm{R}_{C\times HW}$. The result is restructured to produce a tensor of dimension C×H×W by performing matrix multiplication between T_D and the spatial attention matrix T_S.

Finally, add components to the original feature T_A to produce the final output $\mathrm{T}_P \in \mathrm{R}^{C\times H\times W}$, which should mirror the finished representation of the distant backdrop. This is done by multiplying the result matrix of the aforementioned multiplication by the proportional parameter. The entire computation may be stated as follows:

$$T_{Pj} = \delta \sum_{i=1}^{N} \left(T_{Sji} T_{Di}\right) + T_{Aj} \tag{2.7}$$

The channel interest module makes use of self-attention to mannequin the interdependence between channels. By the usage of the interdependence between the function graphs of exceptional channels [15], the function illustration of unique semantics can be expanded to make the community center of attention on some channels with giant weight values.

Use the softmax layer to get the channel interest graph.

$$T_{Xji} = \frac{\exp\left(T_{Ui} \cdot T_{Vj}\right)}{\sum_{i=1}^{C} \exp\left(T_{Ui} \cdot T_{Vj}\right)} \tag{2.8}$$

In addition, T=A is reorganized again to get $\mathrm{T{=}D} \in \mathrm{R}^{C\times H\times W}$, performs matrix multiplication between T_X and T_W, and reorganizes its result $^{RC\times HW}$ into $\mathrm{R}^{C\times H\times W}$. The result is then multiplied by the scale parameter ε and the element summation is performed with T_A to obtain the final output $\mathrm{T}_Q \in \mathrm{R}^{C\times H\times W}$, as shown below.

$$T_{Qj} = \varepsilon \sum_{i=1}^{C} \left(T_{Xji} T_{Wi}\right) + T_{Aj} \tag{2.9}$$

Formula (2.9) indicates that the remaining function of every channel is the weighted sum of the facets of all channels and the unique features, as a consequence setting up a long-term semantic dependency between function maps.

To further strengthen the feature extraction capability of the BERT model, this paper proposes to carry TF-IDP weight on the words after text preprocessing to represent the importance of different word vectors in

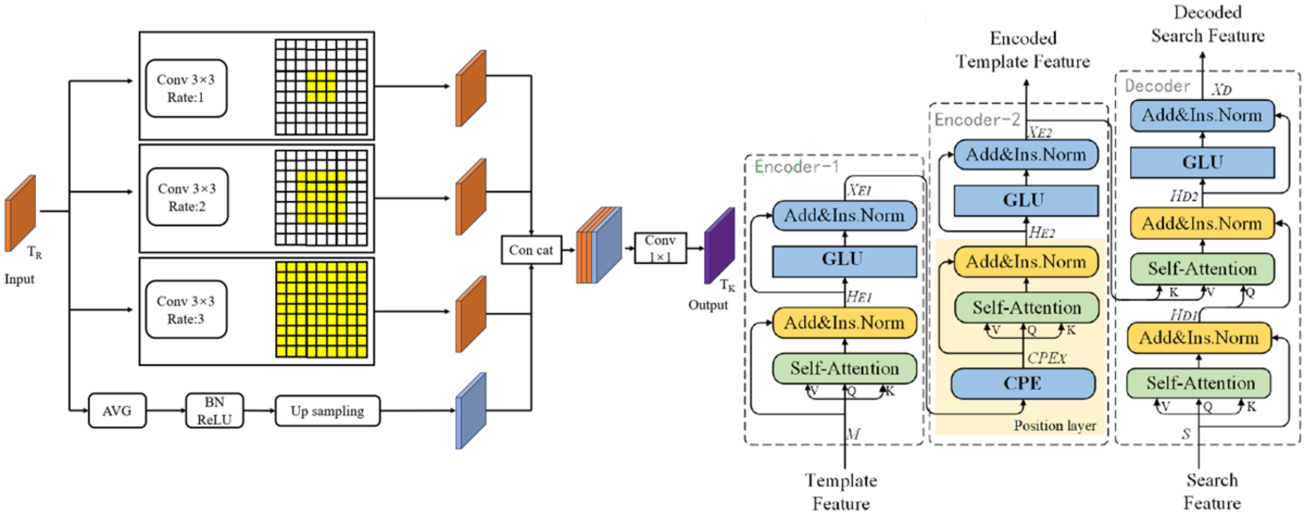


Fig. 2.3: CG-Transformer

the text, to improve the text classification effect. TF-IDF weighted BERT model is here shortened to TF-BERT [16].

TF-IDF is used to indicate the importance of a certain word in an article. If a certain word appears several times in one text and less frequently in other texts, it is considered that the word can be used to distinguish and classify the text. TF is word frequency If a word t appears 5 times in a certain article, the TF value of that word in that article is 5. IDF is the inverse document frequency, its value is determined by the number of occurrences of a certain word in the total document, the fewer the number of articles containing the word t in the total document, that is, the less n, the larger the IDF, the more it indicates that the word t can represent the article. The IDF calculation formula of the word t is shown in 2.9.

$$IDF(t) = \log \frac{N}{n_t}. \tag{2.10}$$

The calculation formula of TF-IDF is shown in 10.

$$K\left(t, D_i\right) = \frac{TF\left(t, D_i\right) \times IDF(t)}{\sqrt{\sum \left[TF\left(t, D_i\right) \times IDF(t)\right]^2}} \tag{2.11}$$

Figure 2.3 shows the CG-Transformer structure. This part consists of two layers of encoders and one layer of decoder. To decrease the computational load of the Transformer structure, the encoder is composed of a single-head self-attention module, gated linear unit, and convolutional role coding module. The decoder consists of a single-head self-attention module and a gated linear unit.

The use of self-attention performs a key function in the whole Transformer structure. First, the encoder and decoder enter template characteristic M and search place characteristic S respectively, and then the matrix seriously changes the enter points to achieve question Q, key K, and fee V as the enter of the self-attention section [17]. Then dot product is used to calculate the similarity between Q and K, and a set of interest weights is obtained after passing Softmax. Finally, the weight is dotted with V to obtain a vector with attention weight. The specific calculation process is shown in the following equation.

$$Attention(Q, K, V) = Softmax\left(\frac{QK^T}{\sqrt{d_k}}\right) V \tag{2.12}$$

3. Design of syntax error correction system based on TF-BERT model.

3.1. Query rewriting model construction based on transformer. Compared with English database query rewriting, the difficulty of Chinese database query rewriting task lies in the word segmentation of the query input sequence. In English natural language processing tasks, word segmentation can be divided according to Spaces. After segmentation, a sequence of sentences is processed into several words with clear meaning. So the result of word segmentation is more accurate. For Chinese tasks, due to Chinese writing habits, punctuation marks are usually used at the end of a sentence to distinguish the text. There is no clear separator between each word, which will inevitably lead to ambiguity or wrong distinction in the process of word segmentation [18]. Taking a piece of data in the TableQA data set as an example, when the sentence ”the highest second-hand real estate price” is divided into three words, ”second-hand”, ”real estate” and ”price”, and ”second-hand real estate price” is a column name in the database table, should be used as a proper term as a whole and not be divided. To reduce the influence of Chinese word segmentation on the model effect. You can use a third-party thesaurus such as jieba HanNLP to further ”load” a custom database column noun library on top of its excellent segmentation capabilities. To obtain the specific lexicon, we need to write a script to collect the column name information of all tables in the database and set the weight of the word according to the data amount corresponding to the column name. The larger the amount of data involved, the higher the word weight corresponding to the column name, and the less easily it is split during word segmentation, to obtain a more accurate encoding. However, from the experimental results, the improvement of the rewriting effect of the third-party lexical segmentation is not obvious.

To solve the problem caused by Chinese word segmentation, BERT’s approach is to use character-level encoding to divide each sentence into word-level granularity. Then it relies on its super-large training corpus to make the relationship between words be represented in the first few layers of parameters of BERT. From this perspective, the task of encoding with BERT does not require additional word segmentation processing. In this way, even if the database column names and other proper nouns are divided, it will not affect the encoding effect. For the English words that may appear in the input, BERT divides the words into more detailed word fragments by the Subword Tokenization method, and the encoding method is more flexible and effective. The advantage of this is that it can solve the problem of unknown words in traditional coding [19].

Unlike textual data, tabular data is distributed in a two-dimensional (2-D) structure. The encoding method requires first converting 2-D table data into linearized 1-D sequence inputs, and then feeding the table data into the downstream language model. The method adopted in this paper is to extract the database table name and column name for each query data, and then flatten these data and concatenate them. In the process of concatenation, the database table name is fixed as the starting information, and all column names are concatenated successively. Wrap, as in formula 3.1.

$$Input_E = [CLS, T, SEP, CLS, C_i, SEP, ..., CLS, C_n, SEP] \tag{3.1}$$

For the encoder to combine table information and query information at the same time, to capture the corresponding information between the table and query, the two parts need to be encoded at the same time. So this article concatenates the query input sequence and the table input sequence, and the final input for the encoder will look like this.

$Input_E = [Input_t, Input_q]$ (3.2)

The standard Transformer decoder structure is also used as the skeleton for the decoder.

After the calculation of 6 layers of stacked attention submodules, the output of the encoder is obtained and then input to the decoder. The decoder will combine the attention calculation results of real SQL and the encoder’s hidden state to obtain multiple calculation results with different probabilities at each position of the result sequence. To obtain the optimal SOL result, it is necessary to combine the bunched search algorithm. Specifically, the algorithm builds a search tree from the starting point of the sequence. Based on the breadth-first strategy, the nodes in the first few probabilities (determined by the width of the cluster) of each layer of the tree are selected as the parent nodes of the next layer of search, and the optimal decoding path is finally selected [20].

During the training phase, the other input to the encoder is a real SQL sequence. In the education phase, the actual label is used as the entry of the subsequent kingdom in the decoding process, that is, it performs the position of Teacher Forcing. The advantages of this method: 1. Prevent the incorrect prediction of the preceding kingdom from inflicting all the subsequent countries to bias the incorrect decoding result, right the prediction of the model, and keep away from similar amplification of blunders in the system of sequence generation. two It radically speeds up the convergence pace of the mannequin and makes the coaching manner of the mannequin extra fast and stable. In this chapter, the BERT pre-training mannequin is used as the enter phrase embedding module at the encoder end. Therefore, to make sure that the exceptional inputs of the mannequin map to the equal phrase vector house and keep away from unknown phrase errors, the equal embedding processing wishes to be carried out on the entry of the decoder side, that is, the enter processing is carried out on the actual SQL first.

$$Input_D = [CLS, SQL, SEP] \tag{3.2}$$

3.2. Design of automatic error correction system.. Aiming at English text content, this paper designs a text feature extraction method based on English translation principles, and then compares and analyzes it with typical English parallel corpora to form a grammar error detection method. Considering that the mutual translation process between the source language and the target language is the same, the word vector parameters between the two can remain uniform [21]. In this paper, the encoder and decoder are used to construct the integrated translation structure. In the actual translation process, the probability calculation formula of the integrated translation bar is.

$$P(BA:\theta) = P\left(\eta v, \eta_1, \eta_2, \cdots, \eta_{m-1} : \theta\right) \tag{3.3}$$

The translation network built based on the encoder can be represented by the mathematical formula.

$$\begin{aligned} &\varpi_\tau = sigmoid\left(H_1 e + \lambda_1 + H_2 s_{\tau-1} + L_1\right) \\ &\sigma_\tau = sigmoid\left(H_1 e + \lambda_1 + H_2 s_{\tau-1} + L_2\right) \\ &\xi_\tau = tans\left(H_1 e + \lambda_1 + H_2 s_{\tau-1} + I\right) \\ &s_\tau = \left(1 - \sigma_\tau\right)\xi_\tau + \sigma_\tau \xi_{\tau-1} \end{aligned} \tag{3.4}$$

Using the above formula to constrain the coding process will lead to errors in English grammatical translation results. To resolve this issue, English grammatical features are extracted using a neural network, and the outputs of the feature extraction are then normalized using the softmax function. Create an algorithm that can automatically detect grammar mistakes. The label for a syntax mistake can be represented as follows in the feature space of results for English translation:

$$F = \arg\max \sum_{r=1}^{Z} 1_\psi \tag{3.5}$$

According to the result of English grammar judgment, the probability of English grammar error is calculated.

$$Q = \frac{\sum_{r=1}^{Z} 1_{\beta(r.0)=1}}{E} \tag{3.6}$$

In the TF-BERT model, the Encoder usually encodes all the data of the supply sentence into a fixed-length context vector c, and then the vector c is steady at some point in the decoding method of the Decoder. With Attention, a mechanism for focusing the model's Attention on the currently translated word, the input to the Decoder is no longer a fixed context vector. Instead, the current context vector is calculated based on the currently translated information [22]. Soft Attention considers all inputs but instead of giving equal weight to each input, it pays more attention to certain inputs. The mixed Attention mechanism, Soft Attention, assigns an attention weight, a probability distribution, to each feature. The context vector of its specific region information can be obtained directly by gravity-weighted summation.

To enhance the automatic error correction effect of the system, the horizontal normalization method is used to process the training samples of the grammar automatic error correction model. In simple terms, it is

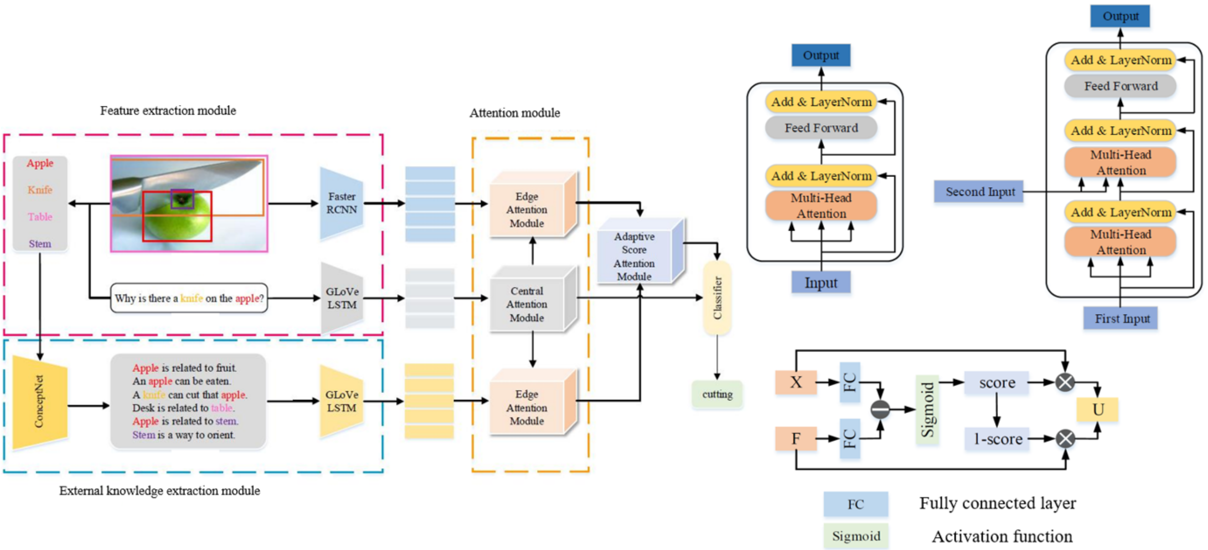


Fig. 3.1: Internal structure based on machine vision

to calculate the input variance of hidden layer neurons, map all neurons into the same space, and calculate the normalized parameters for all hidden layer neurons.

$$\chi = \frac{1}{D}\sum_{u=1}^{D} p_{\mathrm{u}} \tag{3.7}$$

$$\iota = \sqrt{\frac{1}{D}\sum_{u=1}^{D} (p_u - \chi)^2} \tag{3.8}$$

In the formula, χ and l characterize normalized parameters, D represents the range of hidden layer neurons, and u represents hidden layer neurons. The normalized training samples are input into the automatic grammar correction model, and the result of English grammar correction is obtained.

3.3. Attention module design. Specifically, the basic attention unit consists of two major attention modules, namely, the center attention module and the edge attention module. In Figure 3.1, the problem features are first processed by the GloVe word embeddedness and the long/short memory network is obtained through the central attention module. Then, the visual features from the image and the text features from the problem are processed through the edge attention module at the top [23]. In addition, external knowledge is queried through Concept Net, a large public knowledge graph, and the processed knowledge text features are similarly obtained through the bottom-edge attention module. Visual and textual features are given balanced attention through the impact of the adaptive fractional attention module on the problem. Balanced knowledge features, picture features, and semantic features are fed into the classifier to obtain the final answer.

In this paper, the dot product operation between the question and the key is transformed to the product of matrix Q and matrix K, then divided through the scale factor, which is the rectangular root price of dk. The preceding output is then handed to the softmax characteristic and the interest weight is received by multiplying via the matrix V.

$$Attention(Q,K,V) = softmax\left(\frac{QK^T}{\sqrt{d_k}}\right)V \tag{3.9}$$

Each layer of the central attention module is stacked by the basic reduced dot-product attention units of Layer G. The problem characteristic formula it inputs is as follows.

$$Y = [h_1, h_2, \ldots, h_N] \, R^{n \times N} \tag{3.10}$$

This paper combines the attention matrix with the problem features to generate the problem feature Y^g of self-attention, as shown in equation 23. In addition, the problem characteristic of self-attention, Y^g, is also seen as an input to the next basic narrowing dot-product attention unit. This process can be expressed by the formula 3.12 as.

$$Y^g = CAM^g\left(Y^{g-1}\right) \tag{3.11}$$

The side interest module consists of a photo facet interest module and a know-how facet interest module. The photograph part interest module and expertise area interest module have an equal mannequin structure, and every mannequin shape includes G stacked layers, which are composed of two fundamental decreased dot-product interest units. Taking the photograph part interest module as an example, the entry of the picture area interest module comes from the special visible function X cited earlier, and the last output of the hassle characteristic Y^G from the central interest module. The first simple scaling dot-product interest unit in every layer of the photo area interest module, which has the same feature as the single-layer central interest module, inputs X's self-attention for calculating visible points [24].

The 2D scaled dot product interest is linked using the softmax feature to research the interest weight matrix. Thus, the interest matrix is utilized from the final layer of the central interest module to the hassle function of concern, Y^G, to output the problem-based photo feature, X^g, which is the entry to the subsequent layer of scaled dot-product attention. The technique is as follows:

$$X^g = IEAM^g\left(X^{g-1}, Y^G\right) \tag{3.12}$$

Where, the preliminary enter X^o=X, Y^G is the output of the last layer of the central attention module. The output of the final G layer of the image edge attention module is X^G, which represents the picture facets processed via the interest module.

4. Experimental analysis. To verify the effectiveness of this system in detecting and recognizing different combinations of handwritten English grammar errors, the experiment will select the hardware and software parts of the system and set the parameters. Among them, the MU3E200M/C camera with 2 million pixels and a frame rate of 60fps was selected to capture handwritten English character images. All experiments were conducted on a CPU/GPU compute node configured with a xx GHz Intel-Xeon Skylake processor and yy GB RAM. The lens model selected VS-2518VM, equipped with a C-port of 2 million pixels to match the camera. The light source is blue light, bar light source: the sensor is E3CVS1G, and the detection distance is set to 10 mm.

The English version of Bert-base-uncased is selected as the pretraining model. The encoder of the mannequin consists of eight layers, 768 hidden units, and 12 interest heads. The complete reference quantity is a hundred and ten M. Set the most sentence size of this mannequin to 128, batch measurement to 32, optimization characteristic to Adam, getting to know fee to 0.00001, dropout layer parameter to 0.5. The experimental facts had been amassed through the usage of the CoLA dataset and annotated NUCLE and FCE gaining knowledge of corpora.

To improve the quality of data collection, it is proposed that the data should be cleaned before model training. The specific cleaning steps mainly include reprocessing, eliminating empty lines, special symbol processing, length control, text segmentation and numerical operation, and full half-angle conversion.

To deepen the understanding of each model, the hyperparameters of the above models are listed one by one. The L2 regularization parameter of each model is set to 0.00005, and the activation function of the output layer adopts the softmax function. The hyperparameters of each model are shown in Table 4.1 below.

Using the final data set obtained in this paper, the above models are trained and tested respectively, and the loss curves of each model are obtained.

Table 4.1: Hyperparameters of each model

Hyper-parametric Model	Learning rate	Batch-size	Training rounds	Fill in	Optimi-zation function	Network layer number	Loss function	Hidden layer activation function
Model 1	0.00001	8	200	Y	SGD	21	Cross entropy	ReLU
Model 2	0.00001	8	200	Y	Adam	33	Cross entropy	Leaky ReLU
Model 3	0.00001	8	200	Y	Adam	32	Cross entropy	Leaky ReLU
Model 4	0.00001	8	200	Y	Adam	31	Cross entropy	Leaky ReLU
Model 5	0.00001	8	200	Y	Adam	31	Cross entropy	Leaky ReLU
Model 6	0.00001	8	200	Y	Adam	24	Cross entropy	Leaky ReLU
Model 7	0.00001	8	200	Y	Adam	25	Cross entropy	Leaky ReLU
Model 8	0.00001	8	200	Y	Adam	26	Cross entropy	Leaky ReLU
Model 9	0.00001	8	200	Y	Adam	27	Cross entropy	Leaky ReLU
Model 10	0.00001	8	200	Y	Adam	18	Cross entropy	Leaky ReLU
Model 11	0.00001	8	200	Y	Adam	19	Cross entropy	Leaky ReLU

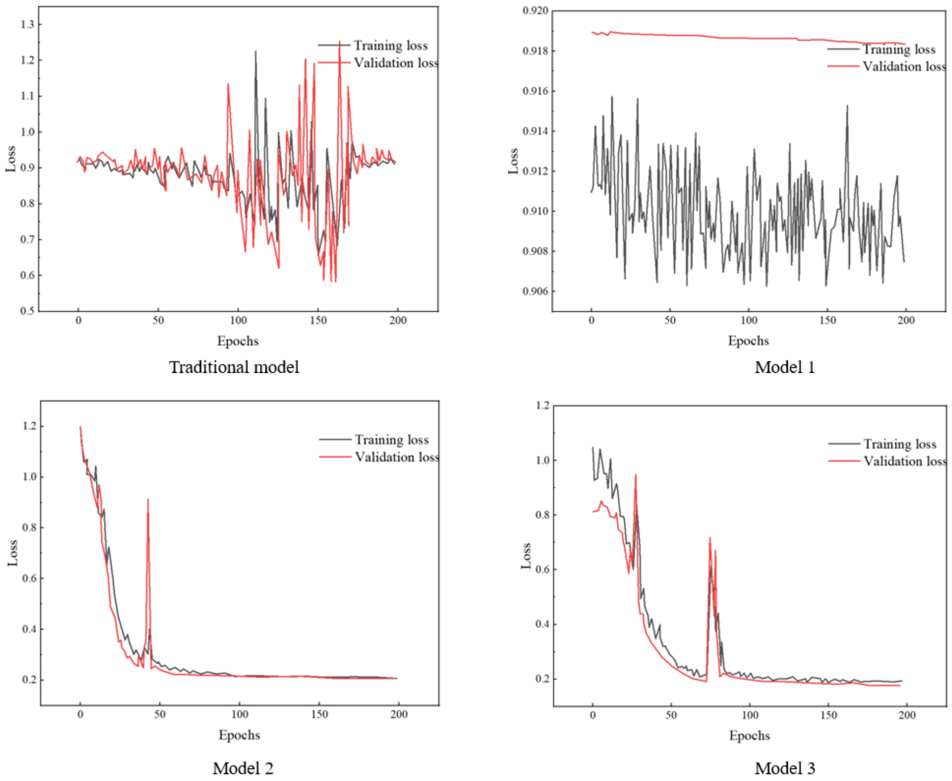
It can be considered from Figure 4.2 that the usual mannequin has apparent oscillations of loss price and accuracy on the coaching set and verification set in this paper, which is no longer steady enough. The loss cost on the take-a-look-at set is 0.7411 with an accuracy of 81.25%. The performance of the traditional model on the data set in this paper is not ideal, and it needs to be further optimized to achieve effective recognition and classification.

As can be seen from Figure 4.1, after reducing the learning rate, the loss value and accuracy of model 1 on the training set still oscillates significantly, while the loss on the verification set gradually decreases, but the change is slow. The loss value on the test set is 0.9157, the classification accuracy is only 43.75%, and the recognition and classification cannot be realized.

Model 2 is the embodiment of the pumpability identification method of mortar based on the 3DCNN+ConvLSTM2D network structure proposed in this paper. It has 3 layers of ConvLSTM2D. Model 2 has optimized and improved model 1, mainly in the following aspects:

1. the optimization function is adjusted to Adam.
2. Adjust the activation function of the hidden layer from ReLU to Leaky ReLU.
3. The networks behind the ConvLSTM2D layer of the third layer are adjusted to :1 GlobalAveragePooling3D layer, 1 Dropout layer, 2 Dense layers, 1 LeakyReLU layer, and 1 Activation layer, respectively. That is, the Global Averagepooling3d-dense-dropout Dense-Activation network structure is used.
4. The learning rate is reduced from 0.005 to 0.00001. The loss of model 3 gradually decreases with the increase of epochs and converges to around 0.17. The loss value of model 3 on the test set is 0.6077, all samples are correctly identified, and the classification accuracy is up to 100%.

As shown in Figure 4.2, Model 4 continues to reduce one ConvLSTM2D layer based on model 3, that is, model 4 has a total of one ConvLSTM2D layer. In addition, the network structure and hyperparameters are consistent with model 3. As can be seen from Figure 4.2, the loss of model 4 gradually decreases with the increase of epochs and converges to around 0.10. The loss value of Model 4 on the test set is 0.4691, and all



Traditional model

Model 1

Model 2

Model 3

Fig. 4.1: Comparison between traditional system and models 1, 2, and 3

samples are correctly identified.

Model 5 tries to improve the learning rate from 0.00001 to 0.0001 based on Model 4. In addition, the network structure and hyperparameters are consistent with model 4. The loss value and accuracy of model 5 on the training set and verification set are oscillated, and cannot converge. All samples, regardless of Pumpability, were identified as Poor Pumpability and could not be effectively classified. The loss value on the test set was 0.8042, and the accuracy rate was only 56.25%.

The loss of model 6 gradually decreases with the increase of epochs and converges around 0.07. The loss value of model 6 on the test set is 0.4153, and the classification accuracy is as high as 93.75%. Compared with model 4, model 6 reduces 7 layers of network, the number of parameters is increased from 1,891,874 to 1,949218, the number of parameters is increased, although the recognition accuracy is slightly decreased, the loss value is reduced, and the convergence speed is greatly improved.

The loss value of model 7 on the test set is 0.3768, and the classification accuracy is as high as 93.75%. Compared with model 6, the learning rate of model 7 is still 0.00001, the number of network layers is increased by one ConvLSTM2D layer, the number of parameters is increased from 1949218 to 2244386, the accuracy rate remains unchanged at 93.75%, the convergence speed is barely improved but the loss value is decreased by 9.30%.

As can be seen from Figure 4.3, the loss of mannequin nine step by step decreases with the amplification of epochs and converges to round 0.05. The loss price of mannequin nine on the check set is 0.4646, and the classification accuracy is up to 93.75%. Compared with mannequin 6, the mastering fee of mannequin nine is nevertheless 0.00001, the range of community layers is decreased by using 7 layers, and the quantity of parameters is expanded from 1949218 to 66725410. The wide variety of parameters is extensively increased, and the accuracy stays unchanged at 93.75%. Although the accuracy on the take-a-look-at set is no longer

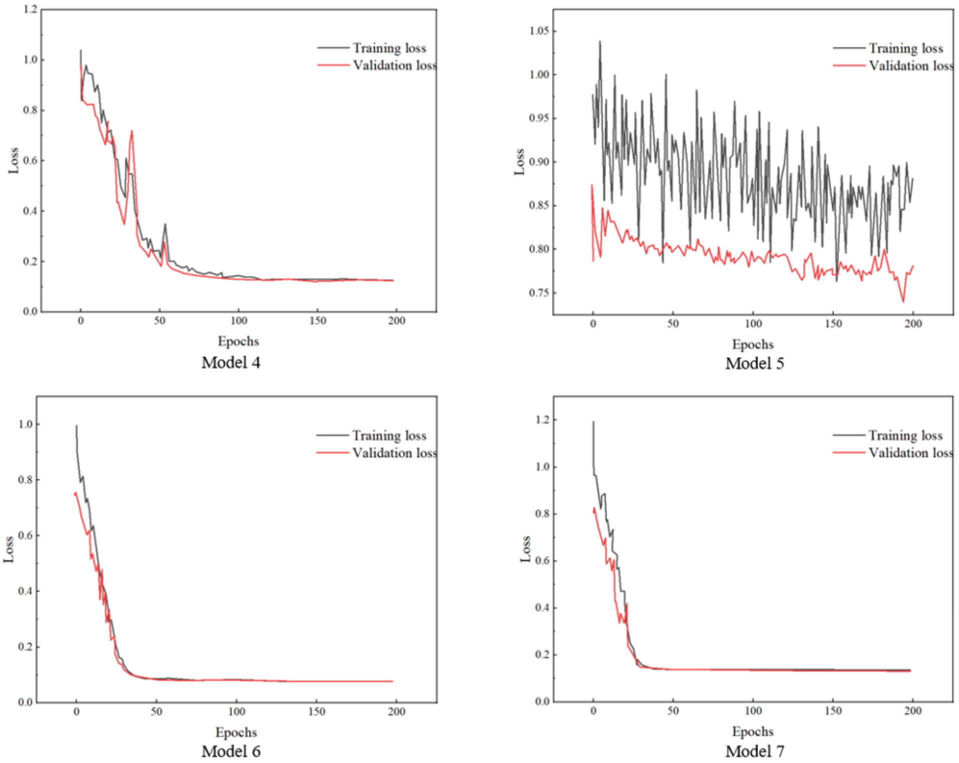


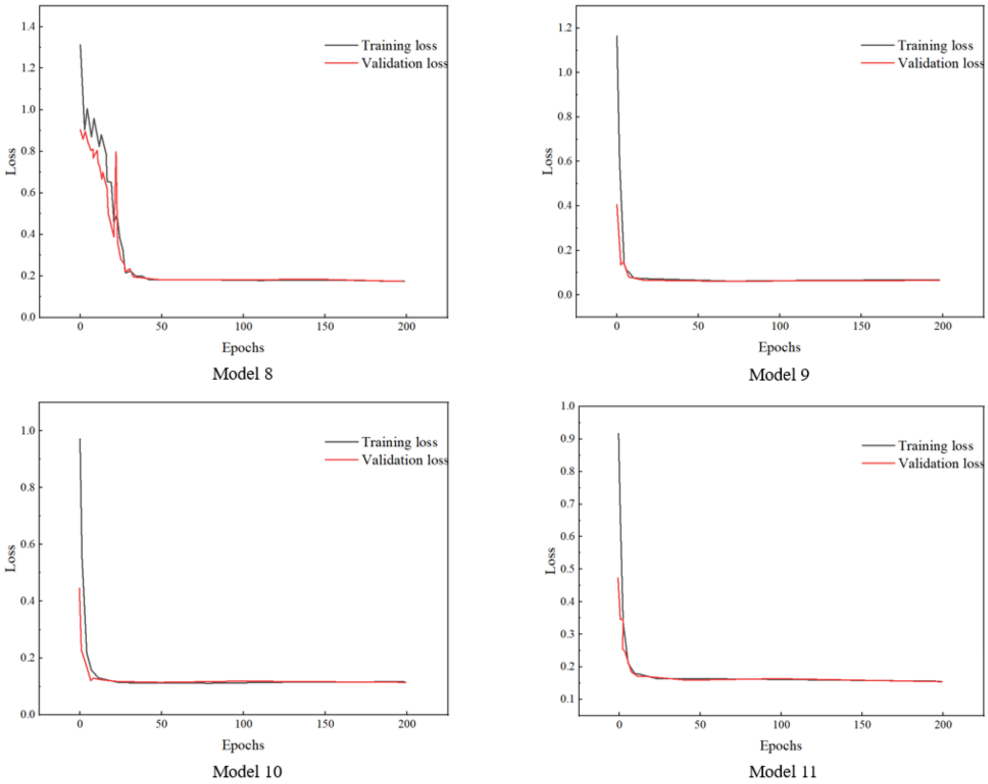
Fig. 4.2: Results of models 4, 5, 6, 7

improved, and the loss cost is barely increased, the convergence velocity is notably accelerated.

Model 11 attempts to add another layer of ConvLSTM2D layer based on model 10, that is, model 11 has 3 layers of ConvLSTM2D layers, in addition, the network structure and hyperparameters are consistent with model 10. In Figure 4.3, the loss of model 11 gradually decreases with the increase of epochs and converges to around 0.15. Model 11 has a loss value of 0.2639 on the test set, and the classification accuracy is up to 100%. Compared with model 10, the learning rate of model 11 is still 0.00001, the number of network layers is increased by one layer, the number of parameters is increased from 67020578 to 67315746, and the accuracy rate is up to 100%. The accuracy of model 11 on the test set is still as high as 100%, and it can realize the correct identification and classification of all samples on the test set. The convergence speed is almost as fast as that of Model 10, and the loss value is reduced by 25.69% compared with Model 10.

5. Conclusion. To detect English translation errors intelligently and efficiently, this paper designs an automatic English translation grammar error detection system based on the optimized BERT machine vision model. A hybrid attention module is introduced into the Transformer model to capture target features in both spatial and channel dimensions and to model the context dependency of target features. Then, the feature maps are sampled by multiple parallel cavity convolutions with different void rates to obtain multi-scale features and enhance local feature representation. Then the input words of the BERT model are weighted by TFIDF to increase the feature extraction capability of the BERT model and construct the TF-BERT model. Specific conclusions are as follows.

1. A visual English automatic translation grammar error detection system based on a mixed attention Transformer is designed. Hybrid attention is embedded in the middle layer of the backbone web to extract more robust features. Image multi-scale feature extraction is realized by using multiple cavity convolution with a smaller cavity rate. The Transformer codec is used to transfer information between



Model 8

Model 9

Model 10

Model 11

Fig. 4.3: Results of models 8, 9, 10 and 11

the template branch and search branch features of the twin network, which improves the accuracy of grammar-translation. The 0-filled convolution method is used to realize more flexible and effective position coding, improve the capability of the web to distinguish similarity, and further increase the efficiency of the automatic translation syntax error detection system.

2. Experiments show that the loss value of the traditional system is as high as 0.7411, the accuracy rate is 81.25%, and the accuracy rate is 75%. The loss of the TF-BERT model increase in this paper decreases with the increase of epochs and converges to about 0.15, with a loss value of 0.2639. When the model in this paper is case 11, the classification accuracy is as high as 100%. The learning rate is still 0.00001, the number of network layers is increased by 1 layer, the number of parameters is increased from 67020578 to 67315746, and the accuracy rate is up to 100%.

First, in the warm-up phase, the Transformmer model has become stable, and there is not much room for optimization at the beginning of confrontation training. Second, the construction of the policy gradient may be unstable, resulting in the generator cannot effectively update parameters according to the results of the discriminator. Therefore, it is necessary to explore more reasonable strategy gradient calculation methods in the future, and select appropriate regular constraints to control the amplitude of each gradient update within a reasonable range, so as to obtain better optimization results.

REFERENCES

[1] Golnabi, H. & Asadpour, A. Design and application of industrial machine vision systems[J]. *Robotics And Computer-Integrated Manufacturing.* **23**, 630-637 (2007)

[2] Robie, A., Seagraves, K., Egnor, S. & Others Machine vision methods for analyzing social interactions[J]. *Journal Of Exper-*

imental Biology. **220**, 25-34 (2017)

[3] Yakui, L., Hongyun, L., Tianzi, L. & Others Research on Mechanical Characteristics Measurement Method of High Voltage Circuit Breaker Based on Machine Vision [J]. *Transactions Of China Electrotechnical Society.* **202432** pp. 35432-20235

[4] Yiquan, W., Lanyue, Z., Yubin, Y. & Others Research status and prospect of PCB defect detection algorithm based on machine vision [J]. *Chinese Journal Of Scientific Instrument.* **43**, 1-17 (2022)

[5] Qihong, Z., Yi, P., Junhao, C. & Others Yarn break location method for yarn joint Robot based on Machine vision [J]. *Acta Textile Sinica.* **43**, 163-169 (2022)

[6] Yong, L., Peijian, S., Qijian, O. & Others Measurement method of irregular shape ultra-thin heat pipe width based on machine vision [J]. *Journal Of South China University Of Technology (Natural Science Edition).* **50** pp. 4 (2022)

[7] Jie, R., Peng, L., Yitong, X. & Others Measurement Method of dynamic lift of hanging string based on Machine vision [J]. (China Railway Science,2022)

[8] Chen, Y., Chao, K. & Kim, M. Machine vision technology for agricultural applications[J]. *Computers And Electronics In Agriculture.* **36**, 173-191 (2002)

[9] Ren, Z., Fang, F., Yan, N. & Others State of the art in defect detection based on machine vision[J]. *International Journal Of Precision Engineering And Manufacturing-Green Technology.* **9**, 661-691 (2022)

[10] Oren, M. & Nayar, S. Generalization of the Lambertian model and implications for machine vision[J]. *International Journal Of Computer Vision.* **14** pp. 227-251 (1995)

[11] Sonka, M., Hlavac, V. & Boyle, R. Image processing, analysis, and machine vision[M]. (Cengage Learning,2014)

[12] Devlin, J., Chang, M., Lee, K. & Others Bert: Pre-training of deep bidirectional transformers for language understanding[J]. (2018), arXiv preprint

[13] Dickmanns, E. & Graefe, V. Dynamic monocular machine vision[J]. *Machine Vision And Applications.* **1**, 223-240 (1988)

[14] Marcel, S. & Rodriguez, Y. Torchvision the machine-vision package of torch[C]//Proceedings of the 18th ACM international conference on Multimedia. *1485-1488.* (2010)

[15] Vision, V. Automated visual inspection and robot vision[M]. (Prentice-Hall,1991)

[16] Penumuru, D., Muthuswamy, S. & Karumbu, P. Identification and classification of materials using machine vision and machine learning in the context of industry 4.0[J]. *Journal Of Intelligent Manufacturing.* **31**, 1229-1241 (2020)

[17] Rehman, T., Mahmud, M., Chang, Y. & Others Current and future applications of statistical machine learning algorithms for agricultural machine vision systems[J]. *Computers And Electronics In Agriculture.* **156** pp. 585-605 (2019)

[18] Kataoka, T., Kaneko, T., Okamoto, H. & Others Crop growth estimation system using machine vision[C]//Proceedings 2003 IEEE/ASME international conference on advanced intelligent mechatronics (AIM 2003). *IEEE.* **2** (2003)

[19] Lenz, R. & Tsai, R. Techniques for calibration of the scale factor and image center for high accuracy 3-D machine vision metrology[J]. *IEEE Transactions On Pattern Analysis And Machine Intelligence.* **10**, 713-720 (1988)

[20] Transforms, P. Properties and machine vision applications[J]. *CVGIP: Graphical Models And Image Processing.* **54**, 56-74 (1992)

[21] Ranft, B. & Stiller, C. The role of machine vision for intelligent vehicles[J]. *IEEE Transactions On Intelligent Vehicles.* **1**, 8-19 (2016)

[22] Kopparapu, S. Lighting design for machine vision application[J]. *Image And Vision Computing.* **24**, 720-726 (2006)

[23] Dickmanns, E. & Graefe, V. Applications of dynamic monocular machine vision[J]. *Machine Vision And Applications.* **1**, 241-261 (1988)

[24] Wildes, R. & Asmuth, J. Green G L, et al. A machine-vision system for iris recognition[J]. *Machine Vision And Applications.* **9** pp. 1-8 (1996)

Edited by: Zhengyi Chai
Special issue on: Data-Driven Optimization Algorithms for Sustainable and Smart City
Received: Oct 31, 2023
Accepted: Mar 25, 2024