applied sciences

MDPI

*Article*

# RQ-OSPTrans: A Semantic Classification Method Based on Transformer That Combines Overall Semantic Perception and "Repeated Questioning" Learning Mechanism

**Yuanjun Tan [1], Quanling Liu [1], Tingting Liu [2], Hai Liu [1,3,4], Shengming Wang [1] and Zengzhao Chen [1,3,*]**

[1] Faculty of Artificial Intelligence in Education, Central China Normal University, Wuhan 430079, China
[2] School of Education, Hubei University, Wuhan 430072, China
[3] National Engineering Research Center for E-Learning, Central China Normal University, Wuhan 430079, China
[4] Shenzhen Research Institute, Central China Normal University, Shenzhen 518051, China
* Correspondence: zzchen@mail.ccnu.edu.cn

**Abstract:** The pre-trained language model based on Transformers possesses exceptional general text-understanding capabilities, empowering it to adeptly manage a variety of tasks. However, the topic classification ability of the pre-trained language model will be seriously affected in the face of long colloquial texts, expressions with similar semantics but completely different expressions, and text errors caused by partial speech recognition. We propose a long-text topic classification method called RQ-OSPTrans to effectively address these challenges. To this end, two parallel learning modules are proposed to learn long texts, namely, the repeat question module and the overall semantic perception module. The overall semantic perception module will conduct average pooling on the semantic embeddings produced by BERT, in addition to multi-layer perceptron learning. The repeat question module will learn the text-embedding matrix, extracting detailed clues for classification based on words as fundamental elements. Comprehensive experiments demonstrate that RQ-OSPTrans can achieve a generalization performance of 98.5% on the Chinese dataset THUCNews. Moreover, RQ-OSPTrans can achieve state-of-the-art performance on the arXiv-10 dataset (84.4%) and has a comparable performance with other state-of-the-art pre-trained models on the AG's News dataset. Finally, the results indicate that our method exhibits a superior performance compared with the baseline methods on small-scale domain-specific datasets by validating RQ-OSPTrans on a specific task scenario by using our custom-built dataset CCIPC.

**Keywords:** topic classification; residual connection; pre-trained model; Transformer



**Citation:** Tan, Y.; Liu, Q.; Liu, T.; Liu, H.; Wang, S.; Chen, Z. RQ-OSPTrans: A Semantic Classification Method Based on Transformer That Combines Overall Semantic Perception and "Repeated Questioning" Learning Mechanism. *Appl. Sci.* **2024**, *14*, 4259. https://doi.org/10.3390/app14104259

Academic Editor: Antonio Fernández-Caballero

Received: 13 April 2024
Revised: 5 May 2024
Accepted: 15 May 2024
Published: 17 May 2024



## 1. Introduction

The volume of textual data has experienced exponential growth with the extensive application of information technology across various industries. These data encompass vast textual information from diverse domains, including recorded speech data, news articles, social media posts, academic papers, and more. Effectively classifying and organizing textual data have become crucial tasks. Text classification facilitates the precise identification of textual information, thereby enhancing text comprehension and enabling the provision of more intelligent and personalized services across diverse application scenarios.

Topic classification, as a pivotal engineering task in natural language processing (NLP), can rapidly and accurately discern the relevant textual theme. This task plays a significant role in various application scenarios, such as educational assessments, sentiment analysis, and public opinion monitoring. Conducting sentiment analysis on social media aids businesses in understanding users' attitudes and emotions toward their products or services, facilitating adjustments in marketing strategies. Additionally, performing topic classification in news reporting assists media organizations in swiftly comprehending the trends of current events, thereby enhancing reporting efficiency.

### *1.1. Challenges*

Although machine-learning and deep-learning methods excel in text classification tasks, their performance may be constrained by the window size and parameter gradients. In some challenging scenarios (Figure 1), such as the same type of text with different contents, transcription errors, and written and spoken expressions with the same meaning, information that plays a crucial role in text classification is hidden in expressions with less similarity. Accordingly, leveraging as many semantic associations as possible from a holistic perspective of full-text data is essential in order to attain robust and highly accurate predictions. In specific application domains, some pressing challenges remain to be addressed. The challenges can be summarized into the following three aspects:

1. The presence of colloquial expressions and noise in real-time interactive texts such as news interviews, teacher lectures, and conference interviews will cause interference in semantic comprehension, and put forward higher requirements for the accuracy of topic classification.
2. The same topic may have various expressions in different contexts, different sub-topics of the same topic may vary greatly, and the text organization structure may show subjectivity, thus increasing the complexity of the topic classification task.
3. Speech recognition systems have achieved excellent text transcription performance in recent years, but, in the face of the situation described in Challenge 1, they may further exacerbate the difficulty of comprehension of colloquial expressions in the spoken table or recognizing the noise as text, which may affect the accuracy of topic classification.

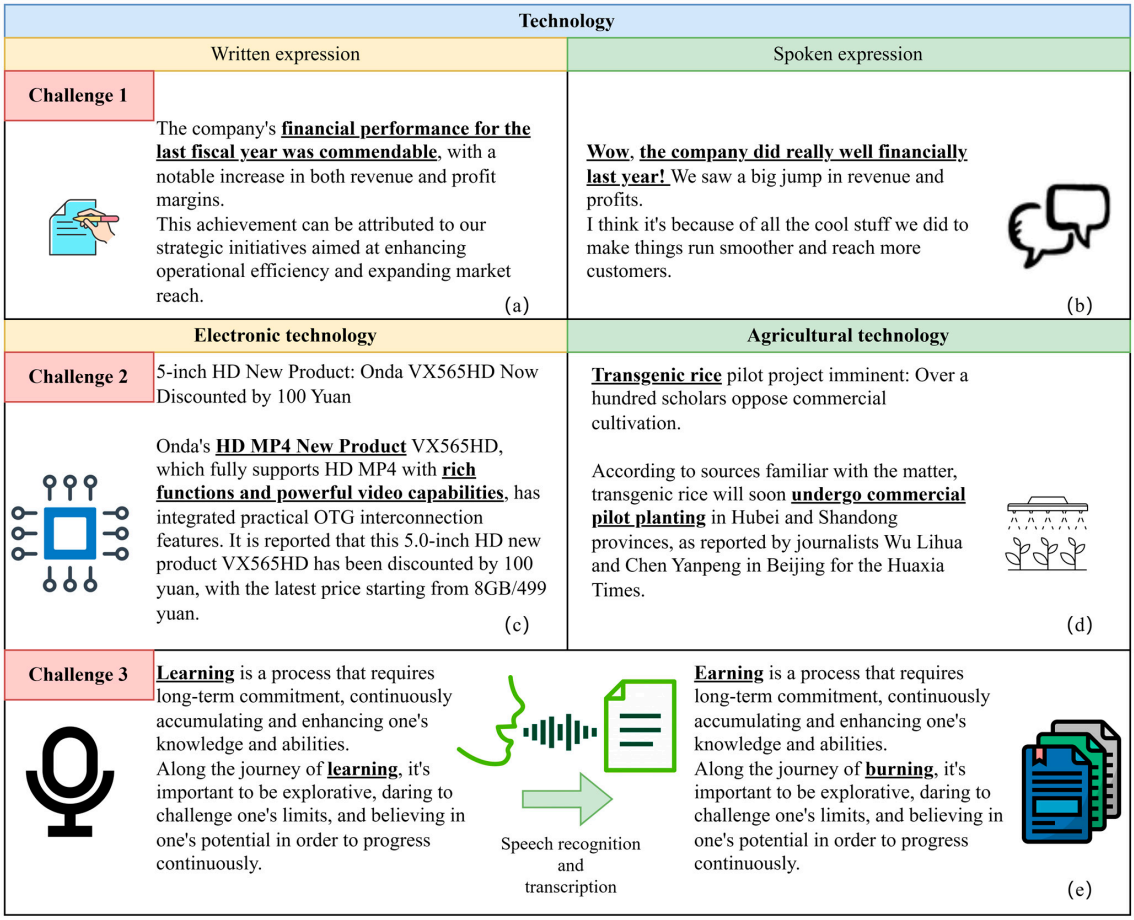



**Figure 1.** Three challenges exist in text classification. (**a**,**b**) Written and spoken expressions with the same meaning. (**a**) Written expression. (**b**) Spoken expression. (**c**,**d**) Two paragraphs on the same topic. (**c**) Electronics technology within the technology category. (**d**) Agricultural technology within the technology category. (**e**) Error in speech recognition transcription.

### 1.2. Observation and Insights

In this study, we identified two insights through observation to facilitate text classification by leveraging the fusion of critical local contextual semantic and global semantic information. First, crucial semantics determining text categories often manifest in different positions across various text genres. For instance, Figure 1a,b depict that the critical information determining the category attribution in news articles typically appears at the beginning of paragraphs. Emphasizing the use of the initial paragraph information aids in determining the text categories. Nonetheless, the distinctions between electronic technology products and agricultural technology applications depicted in Figure 1a,b, respectively, are significant, necessitating a further category differentiation. Figure 1d,e show that the meanings conveyed in written and spoken expressions are roughly the same. In such cases, genre classification based solely on the overall semantics poses significant challenges, as discerning genre categories without a careful comprehension of each sentence becomes difficult. Without integrating full-text semantics with sentence-level semantics to confirm text categories in extensive samples, classification errors are prone to occur. Thus, confirming the approximate classification of texts through holistic semantic learning, combined with fine-grained learning based on sentence-level associations, facilitates the accurate prediction in samples with implicit nested label associations.

Second, when critical text keywords are erroneous or missing, correction can be achieved by leveraging other semantic information within the entire text. This notion suggests that, in long-text learning, individual keywords alone cannot determine the overall semantics. Guiding classification with overall semantics can enhance accuracy. For example, Figure 1c demonstrates that the first word guiding the overall semantics in the original text is incorrectly recognized after being transcribed by a speech recognition system. Upon reading the entire text, the existence of this error becomes apparent and does not result in any difficulty in understanding the semantics.

**Key insight I: Overall semantics guide understanding**

In specific texts, situations arise where critical keywords are missing or erroneous. At such times, the overall semantics of the text are crucial for comprehension and are more robust than word-level and sentence-level semantic learning (Figure 1c). Although various methods exist for text learning, only the overall semantics can guide one's understanding, rectifying errors in word and sentence vectors that may disrupt classification due to internal critical information errors.

**Key insight II: Joint semantic learning is robust**

We find that word vector semantic learning and overall semantic learning have advantages in text feature learning. Accordingly, these two learning modes can serve as parallel structures to assist each other in feature learning. We term this mechanism joint semantic learning. When the sentence-level semantics and overall semantics coexist, extreme situations in text data can be addressed, as depicted in the three parts of Figure 1. In comparison with CNN and RNN architectures, Transformers possess notable long-distance dependency learning capabilities, rendering them suitable carriers for joint semantic learning.

In this section, we define three levels of semantic representation units—"overall semantics of the text", "word-level semantics", and "sentence-level semantics"—and the specific meanings and purpose of these three levels of units will be described in Section 3.

### 1.3. Contributions

In this work, we aim to address the three challenges mentioned above in the task of categorizing topics. Inspired by the above two key insights, we propose a method based on a residual-connected Transformer encoder and an overall semantic perception approach named RQ-OSPTrans. Specifically, this method preserves a copy for the input of each layer of the encoder, maximizing the retention of all features of the text sequence and allowing the model to relearn to increase the stability in classification. The preserved copies serve as a "re-questioning" of the previous round of learning results as the network deepens, requiring

the model to more thoroughly relearn and provide classification predictions. Moreover, with the aid of feature space normalization, this model maintains a stable distribution of features in the input's feature space, enhancing the precision of attention mechanism learning and the determinism of feature partitioning in the multi-layer perceptron (MLP). In the BERT [1] output structure, an overall semantic perception module based on the global semantic pooling expression is applied to learn global semantic features from high-dimensional data and guide the classification output. Finally, a λ factor is introduced to weigh the sum of the classification and guided outputs, combined with Softmax weights, ensuring that the model's output fully considers the capabilities of each learner and the contribution of the different semantics to the classification.

The main contributions of this work can be summarized as follows:

1. This work proposes a method based on a repeat question module and an overall semantic perception approach named RQ-OSPTrans to address the challenges in long-text classification with noise. The model achieves a more effective long-text classification by combining non-adjacent sequence question repetition learning with overall semantic perception.
2. This work establishes a dual-classification learning mechanism through parallel networks to compute the classification and guided outputs separately for annotating labels on a given text segment. A semantic correction mechanism, amplified by Softmax weights, is introduced to enhance the accuracy of identifying textual knowledge points and detecting moral values through learning non-adjacent features in long sequences.
3. RQ-OSPTrans is validated to be competent in Chinese text topic recognition and derived semantic detection in most scenarios through extensive experiments on the self-built datasets CIPCC, Chinese topic recognition datasets, and publicly available English datasets. This mechanism performs at par with state-of-the-art methods in English tasks.

The remaining sections of this paper are structured as follows: Section 2 introduces related work, including deep-learning methods and pre-trained methods based on the Transformer. Section 3 describes the detailed structure and network propagation logic of RQ-OSPTrans, comprising word sequence embedding learning, repeat question modules, and overall semantic perception modules. Section 4 details the dataset selection and training results. Section 5 concludes our work.

## 2. Related Work

This work refers to many text classification methods, and, after systematically observing the methods from traditional methods to pre-trained models, we draw on the computational mechanism of advanced models to build our models.

### *2.1. Traditional Methods for Text Classification*

In previous research, problem transformation methods and adaptive techniques have made a staged progress on this issue. Problem transformation methods, such as binary relevance (BR) [2], classifier chains (CC) [3], and CLR [4], have basic text classification capabilities, address multi-label classification by constructing binary classifiers for each label, sequentially considering label classification, or transforming classification into a ranking problem. Adaptive techniques utilize sophisticated machine-learning methods to directly address associations between labels and imbalanced different classes. Algorithms, such as Rank-SVM [5], address nonlinear issues but may not consider label interdependencies. The CML [6] algorithm combines conditional random fields to handle label correlations in multi-label classification.

Deep-learning methods have emerged as the mainstream approach for text classification compared with question transformation methods and machine-learning approaches. Existing deep-learning models can be categorized into several types: convolutional neural network models (CNN, e.g., TextCNN [7] and DPCNN [8]), recurrent neural networks and

their variants (recurrent neural network [RNN], e.g., RNN [9], LSTM [10], GRU [11], and their bidirectional models), and joint network models (e.g., CNN–RNN [12]). All these models achieved an impressive performance.

### 2.2. Text Classification Method Using Attention Mechanism and Joint Network

Although text classification methods based on CNN or RNN are highly effective, incorporating the knowledge of the document structure into the model architecture can achieve an improved representation. Yang et al. [13] proposed the hierarchical attention network (HAN) in this context. HAN utilizes word-level and sentence-level attention mechanisms. This mechanism enables the model to focus on crucial content within the document, thereby better integrating the context to assign higher weights to keywords and sentences, ultimately resulting in an excellent performance. You et al. [14] introduced the Attention-XML model, which effectively leverages the most relevant multiple labels from an extensive label set to annotate a given text. This approach efficiently addresses the long-tail problem of labels, significantly improving the classification performance of each label. Zeng et al. [15] proposed a multi-task multi-granularity attention network. By combining coarse-grained classifiers and fine-grained classifiers, data with category intersections are effectively learned. At the same time, the gradient control module controls the gradient propagation of the two-level granular learner to suppress the interference caused by irrelevant features, achieving a good classification performance. Related methods [16] using a multi-head attention or connection attention mechanism achieved a great score in recognition tasks.

In addition, the joint network method shows effectiveness in text classification tasks: the architectures of CNNs, RNNs, and attention networks can be combined to form joint networks. In Zhou et al. [17], the CNN was utilized to extract a sequence of higher-level phrase representations, which were subsequently fed into an LSTM to obtain the sentence representation. Lai et al. [18] introduced TextRCNN, building upon the TextCNN model. This model utilizes the RNN mechanism to capture sequential dependencies and CNN to extract crucial local features. Zhou et al. [19] proposed BLSTM-2DCNN based on TextRCNN, which utilizes BLSTM to capture long-term sentence dependencies and 2D convolution with 2D max pooling operations to obtain the representation of the entire sentence.

However, a certain distance limitation will always exist in handling the correlations within text sequences, regardless of whether it is an attention or joint model due to the lack of an effective learning mechanism for long-distance dependencies. The learning capability of the model is always constrained by various factors, such as the window size, preceding hierarchical representation, and model gradients.

### 2.3. Transformer Methods for Text Classification

Vaswani et al. [20] introduced the Transformer model, which made remarkable progress in machine translation and language modeling tasks. This network, utilizing an encoder–decoder structure, captures long-distance dependencies in non-adjacent sequences solely through attention mechanisms. Methods [21,22] based on the Transformer also improved the classification performance on multiple datasets.

Google AI proposed the pre-trained language representation model BERT, which achieved bidirectional context encoding, improving the performance of downstream NLP tasks. In 2019, Google introduced the XLNet [23] model, which combines the advantages of the autoencoder and autoregressive models to further optimize the BERT model. XLNet can utilize a bidirectional context while predicting the current word, addressing some of the inconsistencies in pre-training and fine-tuning observed in BERT, and better capturing bidirectional context information. In the same year, the RoBERTa model proposed by Liu et al. [24] and the ALBERT model proposed by Lan [25] improved the performance of the BERT model by adjusting the pre-training strategies and sharing the cross-layer parameters. In 2020, the MacBERT model introduced by Yiming Cui [26] improved upon

the RoBERTa model by using the masked language model (MLM) as a correction masking strategy. Since then, improved models based on BERT have optimized the performance of text classification tasks in multiple environments: I-BERT [27] uses integer operations to quantify reasoning; Sentence-BERT [28] uses twin and triplet network structures to derive semantically meaningful sentence embeddings, achieving a good performance on sentence-to-sentence regression tasks; BinaryBERT [29] binarizes BERT parameters by quantizing activations, pushing BERT quantization to the limit and reducing the model size by 24 times; CoBERT [30] is flexible by learning a consistent representation of self-attention and integrates knowledge from the Transformer and LSTM to improve overall performance; DistilBERT [31] uses knowledge distillation technology to compress the size of the BERT model, maintaining performance while reducing the number of model parameters; and TinyBERT [32] uses the teacher–student knowledge distillation strategy to compress the BERT model to a smaller size, making the model more suitable for resource-constrained environments. SpanBERT [33] modifies the pre-training task to predict contiguous segments (spans) of text rather than individual words to capture long-distance dependencies. Wang et al. [34] introduced the DeepNet model, which improved the stability of the Transformer and successfully extended it to 1000 layers, surpassing the depth of previous deep Transformers by an order of magnitude. The X-transformer model proposed by Huey-Ing [35] in the same year focused on adjusting the sub-layers of the Transformer. The X-transformer model aims to optimize the overall efficiency of the Transformer and achieve a better performance by reducing the number of model parameters in the encoder and decoder, modifying the Transformer architecture, and shortening the training time.

Sun et al. [36] presented a novel large-language-model (LLM) autoregressive architecture called RetNet. This model introduces a multi-scale preservation mechanism to replace multi-head attention, providing advantages in terms of parallel training, a low inference cost, and an excellent performance. By contrast, other architectures, such as linear Transformer, recurrent network, and Transformer, could only simultaneously possess two out of these three advantages. The unique characteristics of RetNet suggest that it could become a powerful successor to the Transformer. Meanwhile, LLMs based on the Transformer have made significant breakthroughs in text tasks. Models, such as the GPT [37] series, Qwen, and ChatGLM [38], can complete topic classification tasks with minimal prompts.

In general, fine-tuning a large language model requires a large computational cost and a large amount of data, the pre-trained model lacks the accuracy advantage of subdividing tasks for general NLP tasks, and the deep-learning methods based on sequential models, attention mechanisms, and joint networks still cannot handle long-distance non-adjacent text dependencies well, and lack the mechanism to fully learn such classified cues and guide feature extraction with the extracted cues. Combined with the above features and problems, this work uses a pre-trained language model to represent the text, divides the representation results into three-level semantic representations and inputs them into the upper-layer network, uses a “repeated questioning” mechanism to guide the subsequent feature extraction of high-weight clues, and, finally, combines the overall semantic learning and weight amplification to improve the accuracy of topic classification.

## 3. Proposed Method

The pipeline of the RQ-OSPTrans model is presented in Figure 2. The approach leverages the non-adjacent sequence association learning capability of the Transformer encoder to identify the most significant parts in word sequence embeddings contributing to the classification. Additionally, this model breaks through the dependence on hierarchical semantic perception for long-text semantic learning, achieving more effective long-sequence text classification.

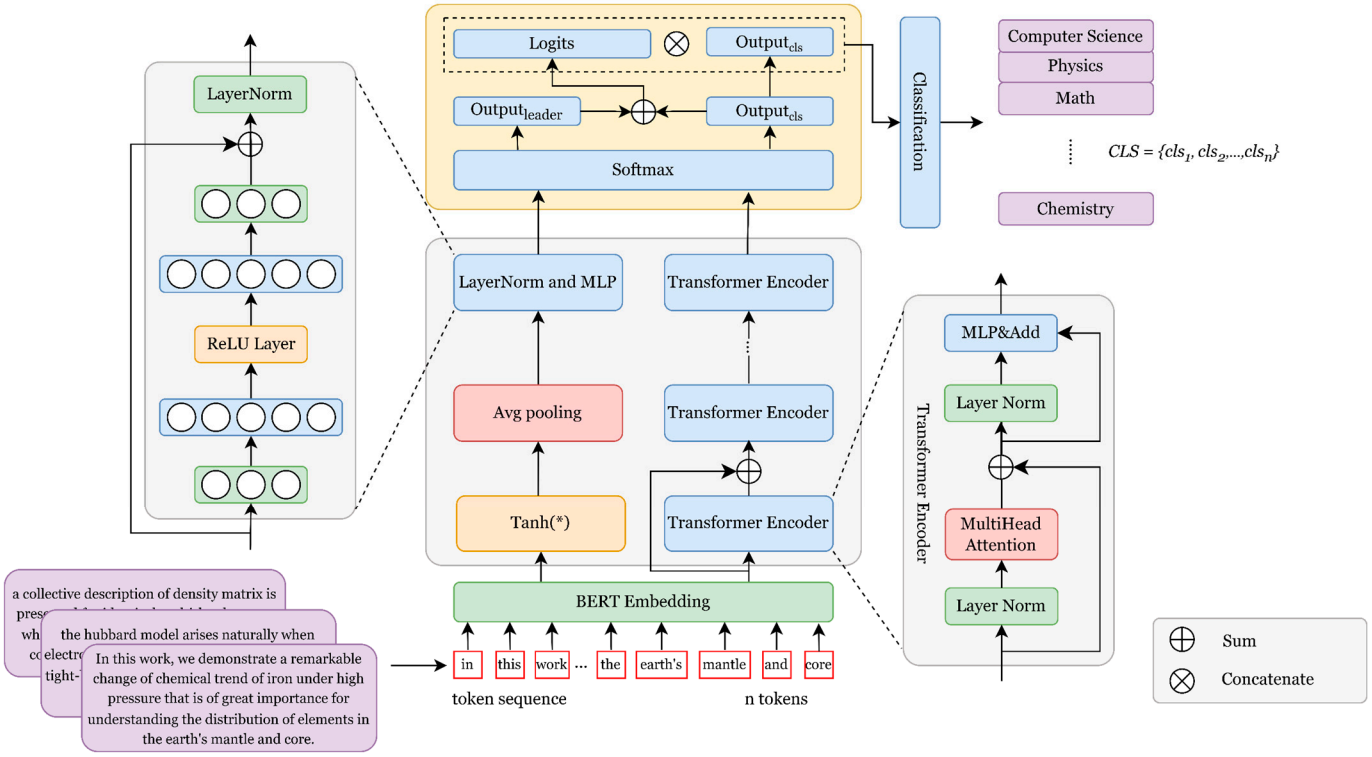



**Figure 2.** RQ-OSPTrans model structure. In the figure, the '*' in "Tanh(*)" represents the weight matrix of the input.

The overall architecture of RQ-OSPTrans consists of three parts: BERT word embedding learning, repeat question learning, and overall semantic perception. With regard to a word sequence S with an arbitrary number of characters $S = \left\{s_1, s_2 \ldots s_{M_{padding}}\right\}$, RQ-OSPTrans inputs these parts into the BERT word embedding learning module to extract sentence feature states $pooler \in R^{M_{embed}}$ and word vector representations $L \in R^{M_{padding} * M_{embed}}$. In a word sequence S, each of them, $s_i$, represents a word-level semantics vector, as well as the elements in $L$. S represents the sentence-level semantics and $pooler$ represents the overall semantics of the text. In other words, the sentence-level semantic representation and the overall semantic representation of each piece of text will be used as model inputs to implement the key insights in Section 1.2, holistic semantic learning guided classification and joint semantic learning, to extract key clues. Thereafter, the repeat question learning module further extracts the implicit associated features of the text, introducing a masking mechanism and residual connections to enhance feature integrity during propagation and obtaining the classification output $Output_{cls}$ of the text sequence. Subsequently, RQ-OSPTrans uses the global feature state pooler, introducing a multi-layer feedforward network with layer normalization to perceive the overall semantics of the sequence, obtaining the guiding output $Output_{leader}$ using ReLU activation. Finally, the weighted average of the two types of outputs is calculated to obtain the final prediction output.

### *3.1. Word Sequence Embedding*

In the word sequence embedding of the text, BERT demonstrates a superior word-level and paragraph-level feature representation performance compared with Glove [39] and Word2Vec [40]. This peculiarity aligns well with the requirements of the word embedding learning module in RQ-OSPTrans, making BERT the chosen core component for this module. After learning through multiple layers of the Transformer encoder, it outputs the last hidden state of the text sequence as the semantic word embedding matrix:

$$\boldsymbol{A} = [\varphi(H_1), \varphi(H_2), \ldots, \varphi(H_n)], \quad (1)$$

where $H_i, i \in [1, 2, 3, \ldots, n]$ represents the hidden state of the word, and $\varphi(H_i)$ denotes the computation function in the BERT model. This model then passes through a linear layer with Tanh activation and uses average pooling to aggregate all word features into a sentence semantic feature vector $pooler_{out}$:

$$x_i = \frac{A_i}{\sum_j^n A_j}, \tag{2}$$

$$pooler_{out} = \frac{1}{M_{padding}}[Tanh(x_1), Tanh(x_2), \ldots, Tanh(x_n)]. \tag{3}$$

Here, we pool each sentence-level semantic representation matrix $\boldsymbol{A}$ evenly in its text length, and fairly consider the semantic representation of each word while ensuring the word embedding dimension, and the semantic capture degree depends on all the dimensions and sentence lengths represented by the word embedding vector of each paragraph, so as to ensure that the overall semantic matrix captures the semantic information of all words. The Tanh activation function is used for nonlinear mapping:

$$Tanh(x) = \frac{e^x - e^{-x}}{e^x + e^{-x}}. \tag{4}$$

We uniformly expand the semantic matrix of word embeddings to the length of $M_{padding}$ to enhance the effectiveness of word vector masking and residual connections. With regard to shorter sentences, we fill the missing parts with zeros. Meanwhile, we truncate the longer sentences to the length of $M_{padding}$. Finally, this module adds positional association features to the semantic matrix using positional information embedding from the Transformer based on the word embedding semantic matrix output by BERT, completing the output of this module.

### 3.2. Repeat Question Module

Similar to the ViT [41] model, we consider a Transformer encoder as a Transformer layer. The repeat question learning module consists of $M_{depth}$ similar layers with residual connections (Figure 3). In the attention part, we utilize nonlinear scaled dot-product attention. The attention score input comprises the key and value from the input. We scale each attention score by the input dimension $M_{embed}$, with the scaling factor being $\frac{1}{\sqrt[2]{M_{embed}}}$. Here, a Tanh activation function is applied to project the attention scores into the nonlinear range of $(-1, 1)$, with the aim of fitting the nonlinear sequence space and separate positive and negative feature weights. This approach helps in reducing the amplification effect of scores for ineffective blocks on the total weight when the input dimension is large, assigning higher weights to effective blocks in the Softmax classification output. The attention score and attention algorithm are shown as follows:

$$AttentionScore(Q, K) = Tanh\left(Q \times K^T\right), \tag{5}$$

$$Attention\ (Q, K, V) = Softmax\left(\frac{AttentionScore(Q, K)}{\sqrt[2]{M_{embed}}}\right) \cdot V. \tag{6}$$

In the first and second layers, an auxiliary masking mechanism is applied for the attention weights of the multi-head attention mechanism. This mechanism helps each attention head to focus on text sequence blocks with high attention scores while minimizing weight waste on ineffective words as much as possible. After obtaining $Q \times K^T$, this mechanism masks elements below the threshold $f = 0.5$ in the attention score matrix $AttentionScore$. Elements with values lower than the threshold $f = 0.5$ are masked to $X_{min} = 10^{-9}$:

$$\forall x_{i,j} \in AttentionScore,\ x_{i,j} = \begin{cases} x_{i,j} & x_{i,j} \geq f \\ 10^{-9} & else \end{cases}. \tag{7}$$

In the subsequent layers, this masking mechanism is removed. The encoder network layers can fully learn from the input while passing a copy of the input to the next layer. The predictions of the current layer are additively fused with the input copy of this layer.

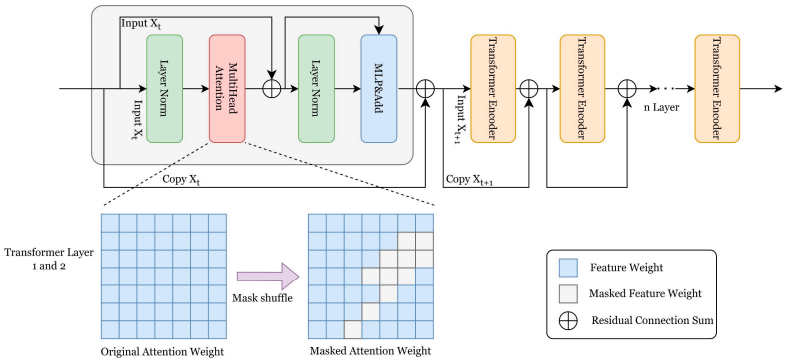



**Figure 3.** Residual connection structure of deep RQ-OSPTrans.

In this work, $M_{embed}$ is projected to dimensions $Query_h$, $Key_h$, and $Value_h$, serving as the input for h attention learners. Nonlinear dot-product scaling attention is applied to each attention learner, allowing the model to focus on feature information in various subspaces of the sequence samples. The granularity of feature extraction is finer compared with a single self-attention mechanism. This module employs a multi-head attention mechanism with h attention heads, where $Query_h = Key_h = Value_h = M_{embed}/h$, requiring the word embedding dimension from Section 3.1 to be evenly distributed across h subspaces. The joint feature learning matrix, denoted as $W_{joint} \in R^{M_{embed} \times M_{padding}}$, is utilized for this purpose. The computation process of the multi-head attention mechanism is as follows. Here, we refer to the output of this part as $Att_{out}$.

$$\begin{gathered} MultiHeadAttention(Q,K,V) = W_{joint}^{T}(concat[head_1, head_2, \ldots, head_h]) \\ where \;\; head_i = Attention(Query_h, Key_h, Value_h)_i. \end{gathered} \tag{8}$$

After the attention network, a scaled MLP with a residual connection to the attention input is introduced. The scaling perceptron maps the output of the attention network to a high-dimensional space of dimension $cls_{dim}$ and performs the first linear partitioning. Thereafter, the dimension of the matrix mapped by the first-dimension increment is flipped and mapped back to the original dimension $M_{embed}$, with a ReLU activation function inserted in the middle of the scaling perception for linear feature correction.

The presence of bias terms in the linear mapping operation affects the feature space distribution of the information. Accordingly, a layer normalization mechanism is established to address this issue. This mechanism calculates the mean and variance of each dimension of the input matrix based on the sample features, stabilizing the data distribution during forward propagation and the gradient during backward propagation. Layer normalization is more suitable for handling long data compared with batch normalization. Normalization focuses on individual sample feature spaces rather than the sentence length and batch size, stabilizing the feature space. The input dimension here is $M_{embed}$, and the dimension after incrementing is $cls_{dim} = 4 \times M_{embed}$. Considering the integrity of the module, we will introduce the computational logic of this module in Section 3.3.

A residual additive connection is utilized in the repeat question learning module (Figure 3). This approach concatenates the input of the previous layer onto the output of the current layer, prompting the Transformer encoder to focus on every element of the input information. During the forward propagation, this mechanism strengthens the perception of the classification results, similar to prompting the model to confirm the classification answer under repeated questioning, reducing the oscillation caused by the randomness of the weight matrix. Assuming that $TransOut_d$ represents the output of the current layer, $Out_d$ represents the output of the original Transformer layer, the model retains a copy of

this output denoted as $TransOut_{d_{copy}}$, while $\lambda$ serves as the guiding factor for the next layer. The formula for the residual connection in the next layer's output is as follows:

$$Out_d = Att_{out} + LayerNorm(MLP(Att_{out})), \tag{9}$$

$$TransOut_{d+1} = Out_{d+1} + \frac{\lambda}{\lambda + \frac{\sum_i \sum_j AttentionScore_{i,j}}{M_{embed}}} TransOut_{d_{copy}}. \tag{10}$$

At the heart of the repetition mechanism lies Equation (10). In a multi-layer Transformer encoder stack, the output of the pre-sequence network layer will affect the learning of the subsequent network, which may lead to a decrease in classification performance since a very small number of classification cues occupy a large attention score and cause the subsequent network to focus only on these parts. Therefore, it is a way to improve the performance by promoting the model to learn more categorical cues and correct the error of the weight allocation of the pre-order network in the subsequent network, and the "repeated questioning" mechanism proposed by us is derived from the attention information contained in the attention score matrix. First, we look at all the attention scores of a text as a whole and sum all the semantically allocated attention values. Secondly, to avoid overfitting the model by focusing on only a few key cues in multiple rounds of questioning, we use the word embedding dimension to smooth the attention value. After that, the output amplification weight is calculated with the help of the guide factor λ to scale the results of the previous round of attention learning, and the additive combination with the results of this round of learning is completed to complete a repeated question learning.

### *3.3. Overall Semantic Perception*

Considering the BERT model output $pooler_{out}$, the Tanh activation function has been introduced, which brings nonlinear mapping. Here, in addition to applying the ReLU activation function between the two scaling layers, another ReLU linear correction is applied after completing the perception machine calculation. This connection retains only positive numbers as guiding elements for the classification output and sets the rest of the elements to zero, achieving the calculation of the MLP output:

$$pooler_{out} = avg\_pooling(Tanh(W \times X + b)), \tag{11}$$

$$MLP_{out1} = ReLU\left(W_1^T \times pooler_{out} + b_1\right), \tag{12}$$

$$MLP_{out2} = LayerNorm\Big(\ ReLU\left(W_2^T \times MLP_{out1} + b_2\right)\ \Big). \tag{13}$$

In each feature sample $X \in R^{M_{embed}}$, the perception matrices are $W_1 \in R^{cls_{dim} \times M_{embed}}$ and $W_2 \in R^{cls_{dim} \times M_{embed}}$. A nonlinear improvement using the Mish activation function [42] is applied in this part to adapt the output feature distribution to the complex feature space of nonlinear activation, mapping the complex feature space to a distribution with minimal gradients. The algorithm for the Mish activation function is as follows:

$$f(x) = x \cdot \mathrm{Tanh}(\varphi(x)), \quad where \quad \varphi(x) = \ln(1 + e^x). \tag{14}$$

In the computation of the multi-layer perceptron, the layer norm calculates the mean $\overline{X_f}$ and variance $\sigma_f$ of the features for each sample. The computation method is as follows:

$$\overline{X}_f = \frac{1}{M_{embed}} \sum_{i=0}^{M_{embed}} x_i, \tag{15}$$

$$\sigma_f = \sqrt[2]{\frac{\sum_{i=0}^{M_{embed}} \left(X_i - \overline{X}_f\right)^2}{M_{embed} - 1}}. \tag{16}$$

The layer norm will normalize the output of the perceptron based on the $\overline{X_f}$ and standard deviation $\sigma_f$:

$$\forall x \in MLP_{out2}^{[i]},\ LN = \frac{x - \overline{X}_f}{\sigma_f}, \tag{17}$$

$$x_{leader} = [LN_1, LN_2, \ldots, LN_n]. \tag{18}$$

The outputs of the learning modules are calculated separately as follows:

$$Output_{cls} = Mish\Big(TransOut_{M_{depth}}\Big), \tag{19}$$

$$Output_{leader} = Mish(x_{leader}). \tag{20}$$

Both outputs are based on the feature space of a sequence of statements as the value set. Normalized feature vectors are calculated based on the feature mean and feature standard deviation.

### *3.4. Softmax Weight Correction and Loss Function*

In our study, two modules are designed to compute the final model classification, handling the processing and loss calculation for the two classification results (Figure 1). During the training phase, the long-sequence repeat question learning module and the overall semantic perception module are treated as two independent classifiers. These modules will compute losses $Loss_{cls}$ and $Loss_{leader}$ for the classification and guidance outputs, respectively. Considering the structure and inputs of the two classifiers, we introduce a guiding factor λ for computing the final output loss. Our output loss function is expressed by the following formula:

$$Loss = Loss_{cls} + \lambda * Loss_{leader}. \tag{21}$$

The design of the loss function aims to leverage the most prominent correlated sequence in the overall semantically reinforced context to contribute to the final classification. These two losses will collaboratively influence the backpropagation process. In either side of the classifier, considering each other's loss serves as the optimization basis to enhance the model's performance.

Similar to the design of the loss function, the final output of the model is a weighted average result from both classifiers. The sum of the weights is not directly used as the denominator here. In the two types of outputs $Output_{cls}$ and $Output_{leader}$, the probabilities for each classification are computed using Softmax. The total classification result of both classifiers is weighted and summed according to the guiding factor. The average operation is then performed across the classifier dimension for the final output *Logits*:

$$Logits = \frac{1}{2} \cdot (softmax(Output_{cls}) + \lambda \cdot softmax(Output_{leader})). \tag{22}$$

Next, this work performs a second averaging operation on $Output_{cls}$ and *Logits* to enhance the contribution of classification learning to the overall classification result and focus on optimizing the long-sequence repeat question feature learning module. This operation effectively combines the second amplification of *Logits* after an amplification by the guiding factor, resulting in the total weight amplification for each feature sample:

$$\forall\ x_i \in Output_{cls},\ \ W_{x_i} = \left( \sum \frac{e^{x_{i,j}}}{\sum_j^{M_{embed}} e^{x_{i,j}}} \right) / \left( \frac{\lambda}{2} + 1 \right). \tag{23}$$

The output after the second amplification is obtained as *Output*:

$$Output = \frac{Logits + Output_{cls}}{2}. \tag{24}$$

## 4. Experiment

The experimental platform is a workstation with Ubuntu 20.04 operating system, equipped with two Nvidia RTX A6000 GPUs, an Intel 13th generation i9 CPU, 96 GB of GPU memory, and 128 GB of RAM. The platform runs PyTorch 2.0.1 deep-learning framework with Cuda 11.7 parallel computing platform, and it is utilized with Python 3.9.

### 4.1. Datasets

With regard to the task of topic recognition in Chinese text classification, this work chooses the THUCNews dataset [43] to evaluate topic identification and detection of long-text features. THUCNews is a news-themed dataset, where most of the texts are long. The utilized version of the dataset for this work includes 10 evenly distributed classification topics, totaling 50,000 news articles covering categories such as real estate, education, technology, stocks, and finance. The samples for each category are sourced from various media outlets on the Internet, and the reporting styles, language logic, and discourse structure for the same topic exhibit significant differences. The texts from live reporting contain a substantial number of colloquial expressions and noise, making it a major challenge to differentiate between topics within complex data combinations. This work selects the AG's News [44] and arXiv-10 [45] datasets for the English text classification tasks. The AG's News dataset comprises four news topics, with 120,000 training samples and 7600 testing samples. The arXiv-10 dataset consists of abstracts and titles from 100,000 scientific papers retrieved from arXiv, covering 10 categories evenly distributed, including subcategories, such as computer science, physics, and mathematics. In our experiments, the dataset is divided into 80,000 training samples and 20,000 testing samples, with each sample's title and abstract combined into a single text. The labels of the two English subject classification datasets were evenly distributed.

On this basis, this work establishes the CIPCC dataset for nurturing elements by extensively collecting classroom discourse texts from the affiliated institution. This dataset is selected from actual classroom teacher lecture materials and exhibits significant features of teaching knowledge points. The language of this dataset is Chinese. Each sample in the dataset includes a specific engineering knowledge point topic. The dataset primarily consists of four classification topics, with over 4000 expert-certified labels for nurturing elements. The biggest challenge of this dataset lies in the uneven distribution of labels. For instance, the "science and innovation literacy" label has 1714 samples, while the "value pursuit" label has 277 samples. The "Humanistic accomplishment" label has 1259 samples and the "Social responsibility" label has 943 samples. Accordingly, we randomly sampled each label using the same proportion to construct the training and testing sets, with a split ratio of 0.8. Detailed information about the CIPCC dataset is described in Table 1; the descriptive information of all datasets selected in this work is summarized in Table 2:

**Table 1.** Information on exemplary nurturing elements from the CIPCC dataset.

| Text ID | Course Category | Label | Text Length |
|---|---|---|---|
| 001 | Engineering | Science and innovation literacy | 152 |
| 002 | Engineering | Science and innovation literacy | 82 |
| 003 | Engineering | Humanistic accomplishment | 54 |
| 004 | Engineering | Social responsibility | 97 |
| 005 | Engineering | Value pursuit | 114 |

**Table 2.** Data statistics settings for comparative experiments.

| Dataset | Domain | Avg-Length | Number of Labels | Label Distribution |
|---|---|---|---|---|
| THUCNews | News | 357 | 10 | uniform |
| AG's News | News | 342 | 4 | uniform |
| arXiv-10 | Academic | 488 | 10 | uniform |
| CCIPC | Education | 117 | 4 | unbalanced |

For CCIPC datasets, we agree on the following abbreviations: "SIL" represents "Science and innovation literacy", "HA" represents "Humanistic accomplishment", "SR" represents "Social responsibility", and "VR" represents "Value pursuit".

### *4.2. Hyperparameter Settings*

The pre-trained BERT model used in this work is sourced from the BERT-base model in the Hugging Face library. (In English dataset, the pre-trained BERT model is at https://huggingface.co/google-bert/bert-base-uncased, accessed on 14 May 2024. In Chinese dataset, the pre-trained BERT model is at https://huggingface.co/google-bert/bert-base-chinese, accessed on 14 May 2024) The hidden layer dimension is set to $M_{embed} = 768$, and $M_{depth} = 12$ Transformer encoder blocks are used with residual connections. The model utilizes a multi-head attention mechanism with $h = 8$. When determining the optimal sentence length, our approach involves assessing the average length of text within the dataset. Adjustments for the Chinese datasets will be implemented to align with this average length. Meanwhile, we will adopt the nearest power of two to the average length for the English datasets for padding. During training, key hyperparameters are carefully tuned to optimize model performance. Initially, the learning rate is set to $5 \times 10^{-4}$, and the cross-entropy loss function is utilized. The learning rate is dynamically adjusted based on the final prediction loss, with a patience of three, facilitating improved convergence toward the optimal solution. Additionally, a dropout probability of 0.5 is consistently applied to the perceptron layer to mitigate overfitting. Gradient optimization uses the Adam optimizer [46], incorporating a weight decay of 0.01 and a warm-up ratio of 0.05. Control over the training process is exerted through the number of epochs, with a maximum tolerance for the loss function's patience set to $epoch\_patience = 4$. Finally, the best model is saved and evaluated on the validation set post-training.

### *4.3. Comparative Methods*

In this work, we selected several advanced pre-trained models and deep-learning models that have achieved excellent results in topic classification to demonstrate that our method can outperform and achieve better performance than these two types of models. All pre-trained models are from https://huggingface.co/, accessed on 14 May 2024.

- **HAN** [13]: Encodes through the three-level composition in the document and finally performs weighted averaging on the sentence-level representations to obtain the representation of the entire document. This method aggregates multi-layer features through weighted aggregation.
- **SHGCN** [47]: The model extracts entities and word nodes through the graph network, uses BERT to learn word vector representation, and uses BiLSTM to strengthen word-level weights, and combines this weight with document features to predict sample classification. Related methods [48,49] also improve classification performance on multiple datasets.
- **KFE-CNN** [50]: The vector representation of text is enhanced by expanding the semantic information of key features and converted into binary vectors for input into the CNN model, which effectively compresses the size of the model while improving the interpretability of the model, while achieving good performance.
- **BTCDC** [51]: Use CNN to obtain semantic features at different levels under the full-text test drive and combine the attention mechanism to strengthen the weight of local features, and then fuse global features and local features to achieve good classification performance.
- **BERT** [1]: Generates word representations through bidirectional context encoding, is applicable to various NLP tasks, and combines encoder attention scores to guide understanding in natural language understanding.

- **ALBERT** [25]: An improvement on BERT that reduces the parameter count, enhances the training efficiency, and achieves good text-understanding capabilities through parameter sharing and cross-layer parameter sharing to reduce model size.
- **RoBERTa** [24]: An optimized version of BERT that improves performance through a larger dataset and longer training time. This model removes the NSP task and achieves good performance with larger batches and longer sequence lengths.
- **XLNet** [23]: Uses a PLM to integrate the advantages of regression language model and autoencoding language model methods, demonstrating excellent ability to capture document-level context correlation information.
- **MacBERT** [26]: Improves BERT with a dynamic masking mechanism, generating a new masking pattern for each input sequence. This model applies a full-word masking strategy in Chinese tasks, achieving better performance than BERT.
- **RBT6** [26]: Adopts a training strategy consistent with RoBERTa but retains fewer Transformer layers. This model achieves text understanding with fewer parameters and a shorter model compared with large pre-trained models.
- **ERNIE** [52]: A knowledge-enhanced pre-trained model that integrates knowledge graphs and text information. This model enhances representations with entity and relationship embeddings, demonstrating outstanding performance in tasks involving domain knowledge.
- **ELECTRA** [53]: Has some similarities with generative adversarial networks. The generator is a smaller MLM model with the goal of recovering masked words. The discriminator only outputs a binary label indicating "replacement" without specifying the actual word, making training more efficient. After pre-training, this work retains only the discriminator for fine-tuning downstream tasks.
- **GPT-2** [54]: A deep-learning model based on the Transformer architecture. This model can directly learn feature representations from raw text through end-to-end learning, avoiding cumbersome feature engineering. The multi-head attention mechanism and large-scale self-supervised pre-training of the model enable it to effectively capture semantic information and quickly adapt to different domains and tasks through fine-tuning, demonstrating outstanding performance and flexibility in text classification tasks.

### *4.4. Experimental Results and Analysis*

The performance data of the baseline model in the comparison experiment were directly derived from the corresponding references. To reflect the real performance, we fine-tuned the model on the datasets to which the baseline model belonged and recorded the fine-tuned classification performance on those datasets that were not experimented with. The performance of the RQ-OSPTrans model on the four datasets is the mean of the 5-fold cross-validation results of the three random initialization parameters to ensure that the model performance is not affected by the random initialization parameters. In the following sections, we will use accuracy metrics to measure the performance of the model. The best performance and the corresponding model name are indicated in **bold**.

#### 4.4.1. Result on the THUCNews Dataset

This work conducts experiments on the THUCNews dataset using a comparative approach, and the experimental results are shown in Table 3. The results demonstrate that RQ-OSPTrans achieves a performance improvement of 0.1% on the test set and 0.4% on the validation set compared with the baseline pre-trained model with the best performance on the test set. Further performance improvement becomes significantly challenging due to the dataset already achieving high accuracy. In comparison with the BERT word vector loader used in this work, the performance improvement on the test set is 0.1% and 1.1% on the validation set, indicating that the method proposed in this work optimizes the output of BERT, achieving superior performance in Chinese tasks. In contrast to the BERT word embedding learning module, RoBERTa, MacBERT, and RBT6 achieve better generalization performance on the validation set, demonstrating that the mask mechanism

adapted to input significantly improves performance. Meanwhile, the BiLSTM structure of the hierarchical attention network does not exhibit superior performance to the Transformer encoder on this task, falling short of the bidirectional attention sequence learning structure in BERT, proving that stacked Transformer encoder modules have significant advantages in long-text learning. RQ-OSPTrans achieves a generalization accuracy of 98.5% compared to previous models on this dataset, indicating that the residual-connected Transformer encoder structure can fully learn the parts of the text that make significant contributions to classifying topics, capturing long-distance semantic correlations.

**Table 3.** Performance comparison of RQ-OSPTrans and state-of-the-art pre-trained models on THUC-News dataset.

| Models | $M_{padding}$ | Test | Dev |
|---|---|---|---|
| HAN [13] | 448 | 88.7 | 90.1 |
| BERT [1] | | 97.6 | 97.4 |
| ALBERT [25] | | 97.6 | 97.9 |
| RoBERTa [24] | | 97.5 | 97.9 |
| RBT6 [26] | | 97.3 | 98.1 |
| ERNIE [52] | | 96.8 | 98.0 |
| GPT-2 [54] | | 97.4 | 97.5 |
| ELECTRA [53] | | 97.5 | 97.9 |
| MacBERT [26] | | 97.5 | 98.0 |
| XLNet [23] | | 97.0 | 97.4 |
| KFE-CNN [50] | | 97.8 | 97.8 |
| BTCDC [51] | | 95.2 | 95.2 |
| **RQ-OSPTrans (ours)** | | **97.7** | **98.5** |

#### 4.4.2. Result on the AG's News Dataset

We also conduct comparative experiments with baseline pre-trained models on the AG's News dataset, and the results are shown in Table 4. Considering the length of the dataset text, the tokenizer's expansion length is set to 256 here to avoid numerous zero elements in the word vectors. When changing the language of the topic recognition task, our method still achieves an accuracy of 94.8%. ELECTRA improves the performance of the discriminator during iterative "replacement" by recovering words from the full-scale scale during training for discrimination, enabling it to identify correct semantics and achieve excellent performance. This approach is similar to the "repeated questioning" mechanism in our method on the parameter level, proving the importance of key semantic weight information in improving the accuracy of long-text classification. Our method fully learns key semantic weight in multi-layer residual stacking, demonstrating the effectiveness of this approach. In comparison with BERT series models, our method achieves a minimum performance improvement of 1.2% in accuracy and a 2.5% improvement over BERT, demonstrating that our two parallel learning modules significantly optimize pre-training weights for this task, effectively learning and extracting weight features in the bidirectional attention mechanism, and demonstrating higher performance in long-term dependency semantic correlation. Overall, RQ-OSPTrans performs exceptionally well among all methods listed in Table 4, demonstrating that our model can further learn implicit semantic information and subtle discriminative representations in long texts based on pre-trained models.

**Table 4.** Performance comparison of RQ-OSPTrans and state-of-the-art pre-trained models on the AG's News dataset.

| Models | $M_{padding}$ | Test |
|---|---|---|
| HAN [13] | 256 | 91.1 |
| BERT [1] | | 92.3 |
| ALBERT [25] | | 90.3 |
| RoBERTa [24] | | 92.3 |
| ELECTRA [53] | | 94.3 |
| GPT-2 [54] | | 93.5 |
| RBT6 [26] | | 93.2 |
| MacBERT [26] | | 93.6 |
| SHGCN [47] | | 88.3 |
| **RQ-OSPTrans (ours)** | | **94.8** |

#### 4.4.3. Result on the arXiv-10 Dataset

Accordingly, RQ-OSPTrans is compared with the baseline models on the arXiv-10 dataset, which has text lengths similar to those of the THUCNews dataset. The results are shown in Table 5. Here, the tokenizer's expansion length is set to 512. On the non-pre-trained models, RQ-OSPTrans exhibits a significant performance improvement of 5.0% compared with the self-learning Transformer framework Protoformer [45] on this task. In Transformer pre-trained models, BERT's bidirectional attention mechanism shows a significant advantage on this task, indicating its excellent capability to learn long-term dependency semantic relationships within sequences of length 512. Nevertheless, our RQ-OSPTrans still achieves a performance improvement of 1.0% on the test set. A possible explanation is that the language expression in the arXiv-10 dataset is more rigorous compared with that in THUCNews, where simpler models may already sufficiently learn semantic features. By contrast, RQ-OSPTrans exhibits sensitivity to the learned sequence length. We will conduct an ablation study on sequence length in Section 4.5.

**Table 5.** Performance comparison of RQ-OSPTrans and the baseline models on the arXiv-10 dataset.

| Model | $M_{padding}$ | Test |
|---|---|---|
| HAN [13] | 512 | 74.6 |
| BERT [1] | | 83.4 |
| ALBERT [25] | | 80.5 |
| RoBERTa [24] | | 77.9 |
| ELECTRA [53] | | 83.2 |
| GPT-2 [54] | | 82.9 |
| RBT6 [26] | | 79.5 |
| MacBERT [26] | | 81.7 |
| Protoformer [45] | | 79.4 |
| **RQ-OSPTrans (ours)** | | **84.4** |

#### 4.4.4. Result on the CCIPC Dataset

Comparative experiments were conducted based on the CCIPC dataset proposed in this work. The results are presented in Table 6. The experimental results show that RQ-OSPTrans achieves an accuracy of 82.7% on the test set, significantly higher than those of other models. RQ-OSPTrans demonstrates better performance compared with the traditional HAN model, indicating the advantages of the pre-trained language models. BERT and ALBERT, as representative pre-trained language models, also achieved decent performance on this task, but 1.5% and 3.8% gaps remain compared with RQ-OSPTrans. Models, such as RoBERTa, RBT6, and ERNIE, exhibit relatively lower accuracy on the test set, which may be due to their insufficient capability to learn semantic information from text containing numerous colloquial expressions and noise. ELECTRA and MacBERT still

fall short of RQ-OSPTrans. By contrast, XLNet performs the worst on the test set. Our RQ-OSPTrans model is better able to capture semantic information from text with the introduction of a residual-connected Transformer structure and further enhances performance on various pre-trained language models. Overall, the experimental results further validate the effectiveness and superiority of our proposed RQ-OSPTrans model in text topic recognition tasks. RQ-OSPTrans not only significantly outperforms other pre-trained language models in terms of performance but also demonstrates strong generalization capabilities, making it suitable for various types of text data.

**Table 6.** Performance comparison of RQ-OSPTrans and the baseline models on the CCIPC dataset.

| Model | $M_{padding}$ | Test |
|---|---|---|
| HAN [13] | 150 | 79.9 |
| BERT [1] | | 81.2 |
| ALBERT [25] | | 78.9 |
| RoBERTa [24] | | 63.8 |
| RBT6 [26] | | 77.8 |
| ERNIE [52] | | 69.5 |
| ELECTRA [53] | | 79.5 |
| GPT-2 [54] | | 77.8 |
| MacBERT [26] | | 78.9 |
| XLNet [19] | | 60.6 |
| **RQ-OSPTrans (ours)** | | **82.7** |

Compared with other comparison experiments, the label distribution of the CCIPC dataset is uneven. To better illustrate the improvement of RQ-OSPTrans over the baseline model, we compare the classification performance of the top five performing models in Table 6 on each label in this dataset, and the results are shown in Table 7. Here, we populate the header with an abbreviated representation of the dataset label. Experimental results show that the accuracy of our proposed method in the four categories is higher than that of other baseline models. From the overall analysis of the experimental data, the classification performance on the "SIL" label is the best, while the classification performance on the "VP" label is the worst. The direct cause of this result is the uneven distribution of labels: the "SIL" label has more than 1700 pieces of data, while the "VP" label has only 277 samples, and this huge difference in data volume makes it easier for the model parameters to fit the classification cues and weights of the "SIL" label, and the classification cues of the "VP" label may be ignored as unimportant information in repeated questions or masked in the auxiliary masking mechanism. In this case, one possible explanation for the performance improvement of the model is that the overall semantic awareness module does not reduce the pooling weight of the text due to the lack of data volume, so that the module can still learn the semantic information of the data with a small sample size, and can still improve the performance of the pre-trained model with an accuracy rate of 1–8%.

**Table 7.** The classification performance of the top five CCIPC accuracy models on each category label of the dataset.

| Model | Rank | SIL | HA | SR | VP |
|---|---|---|---|---|---|
| **RQ-OSPTrans (ours)** | **1** | **0.95** | **0.66** | **0.50** | **0.48** |
| BERT [1] | 2 | 0.95 | 0.64 | 0.50 | 0.45 |
| HAN [13] | 3 | 0.94 | 0.63 | 0.49 | 0.45 |
| ELECTRA [53] | 4 | 0.93 | 0.63 | 0.49 | 0.45 |
| MacBERT [26] | 5 | 0.94 | 0.62 | 0.48 | 0.37 |
| ALBERT [25] | 5 | 0.92 | 0.64 | 0.45 | 0.40 |

### *4.5. Confusion Matrix Analysis*

The confusion matrix reflects the extent to which an algorithm incorrectly predicts similar categories. In this experiment, three datasets, namely, THUCNews, AG's News, and arXiv-10, were selected to evaluate our method. The values on the main diagonal of the confusion matrix represent the prediction probability in the corresponding category (Figure 4). The darker the color, the higher the prediction probability. The true labels are on the *y*-axis, and the predicted labels are on the *x*-axis. We select the top three performance models on each dataset to draw the confusion matrix. The performance of the RQ-OSPTrans method on the corresponding dataset is located at the first of each row (Figure 4a,d,g). The baseline model will show higher prediction confusion on the Society and Education classes in the THUCNews dataset compared with our method. On the AG's News dataset, the confused categories are concentrated in "Business" and "Technology" categories. The RQ-OSPTrans's prediction confusion level in these two categories is equivalent to ELECTRA and better than MacBERT. The performance of the top three performance models on the arXiv-10 dataset is relatively close. In the two categories of "cs" and "eess", similar prediction confusion appears on the class labels. However, the prediction confusion of this method on the "quant-ph" and "stat" categories is better than the baseline model, and RQ-OSPTrans also has better classification accuracy on the "hep-th" category.

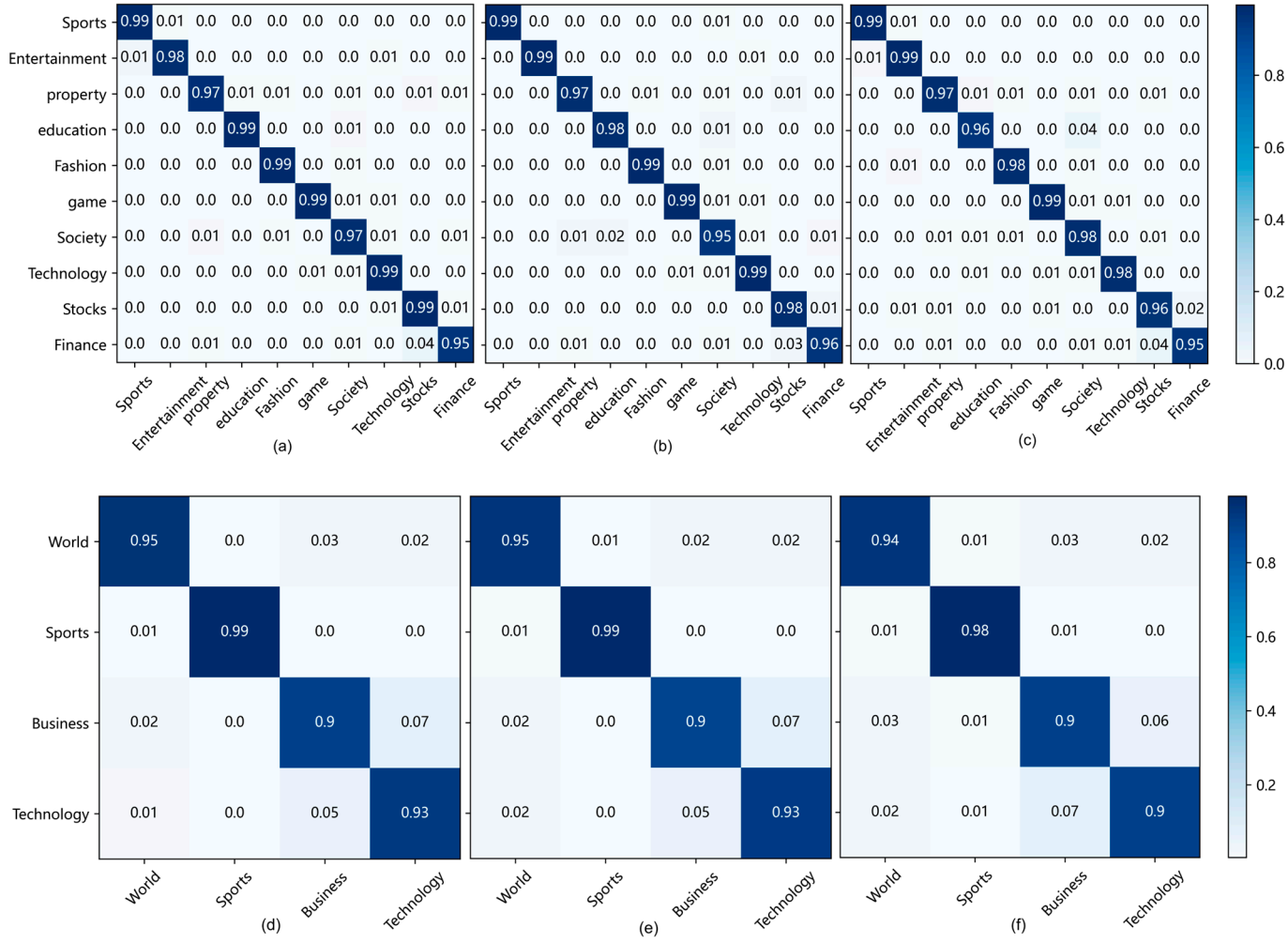



**Figure 4.** *Cont.*

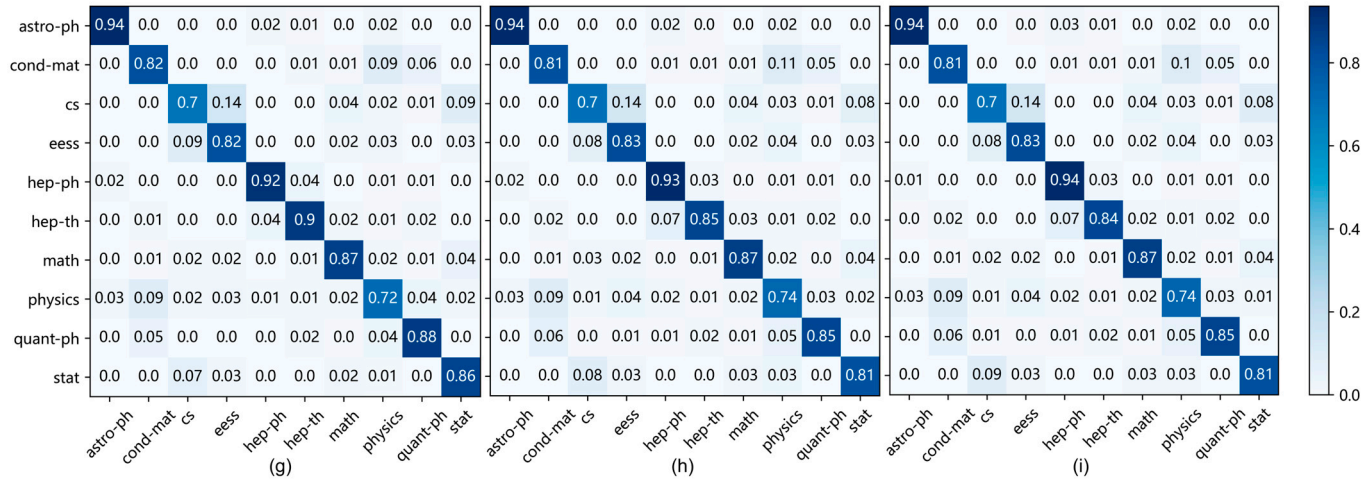



**Figure 4.** Confusion matrix diagram of the top three models in THUCNews, AG's News, and arXiv-10 dataset performance. (**a**–**c**) Confusion matrix diagram of the top three model performance in the THUCNews dataset. (**a**) RQ-OSPTrans in the THUCNews dataset confusion matrix. (**b**) Confusion matrix of RBT6 in the THUCNews dataset. (**c**) Confusion matrix of MacBERT in the THUCNews dataset. (**d**–**f**) Confusion matrix diagram of the top three model performance in the AG's News dataset. (**d**) Confusion matrix of RQ-OSPTrans in the AG's News dataset. (**e**) Confusion matrix of ELECTRA in the AG's News dataset. (**f**) Confusion matrix of GPT-2 in the AG's News dataset. (**g**–**i**) Confusion matrix diagram of the top three model performance in the arXiv-10 dataset. (**g**) Confusion matrix of RQ-OSPTrans in the arXiv-10 dataset. (**h**) Confusion matrix of BERT in the arXiv-10 dataset. (**i**) Confusion matrix of ELECTRA in the arXiv-10 dataset. astro-ph: Astrophysics. cond-mat: Condensed matter. cs: Computer Science. eess: Electrical engineering and systems science. hep-ph: High-energy physics—phenomenology. hep-th: High-energy physics—theory. math: Mathematics. physics: Physics. quant-ph: Quantum physics. stat: Statistics.

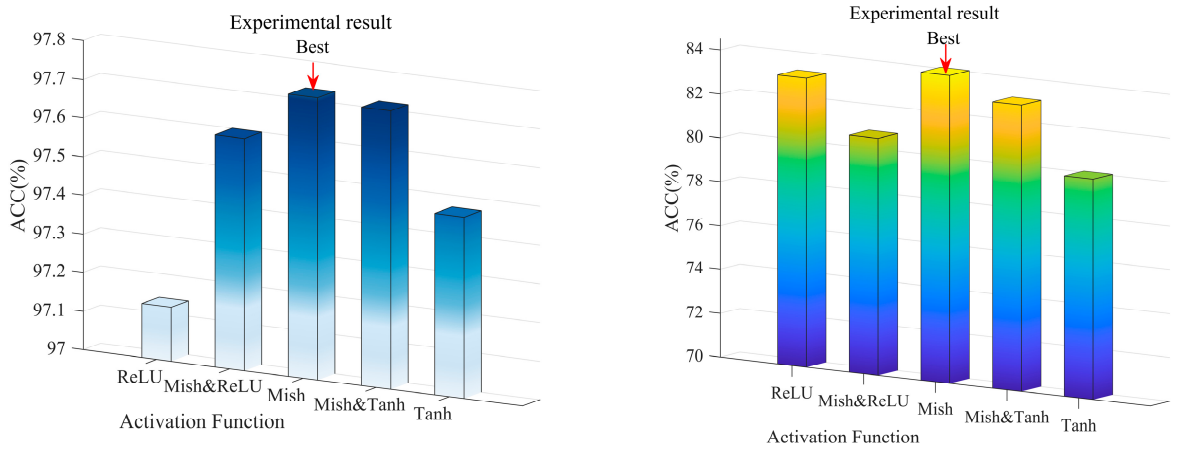
### *4.6. Ablation Study*

We conducted an ablation study on RQ-OSPTrans, focusing on the specific task carried out on the THUCNews dataset.

(1) **Influence of the activation function of the feature output layer on the output classification:**

The output layer activation function is the first layer in the network to participate in backpropagation. The choice of function determines whether complex features can be mapped to a specific space and divided. In this study, experiments were conducted on the THUCNews dataset by using different prototypes and compositions of activation functions combined with Mish activation. The experimental objects include ReLU, Tanh, Mish, Mish&Tanh, and Mish&ReLU functions (functions on both sides of "&" activate $Output_{cls}$ and $Output_{leader}$; no "&" indicates the same mapping for both output types).

The experimental results are shown in Figure 5. In the third section, we introduced the sources of the two types of outputs. The learning process to obtain $Output_{cls}$ is more complex compared with $Output_{leader}$. The Mish function can balance the distribution differences between the two output types while retaining nonlinearity, demonstrating the best performance. By contrast, ReLU can only retain positive weights, neglecting negative ones, resulting in ineffective attenuation of irrelevant information during output weight amplification and suboptimal performance, highlighting the importance of nonlinear mapping in the output. However, excessive nonlinear activations will bring greater classification difficulty. The result of the comparison of the classification performance of Mish, Mish&Tanh, and Tanh activation functions indicates that the accuracy of Tanh activation for $Output_{cls}$ is lower than that using the Mish activation function, indicating that balanced nonlinear mapping results in superior performance. Therefore, RQ-OSPTrans determines the structure of using the same nonlinear mapping for both output types as the optimal choice.

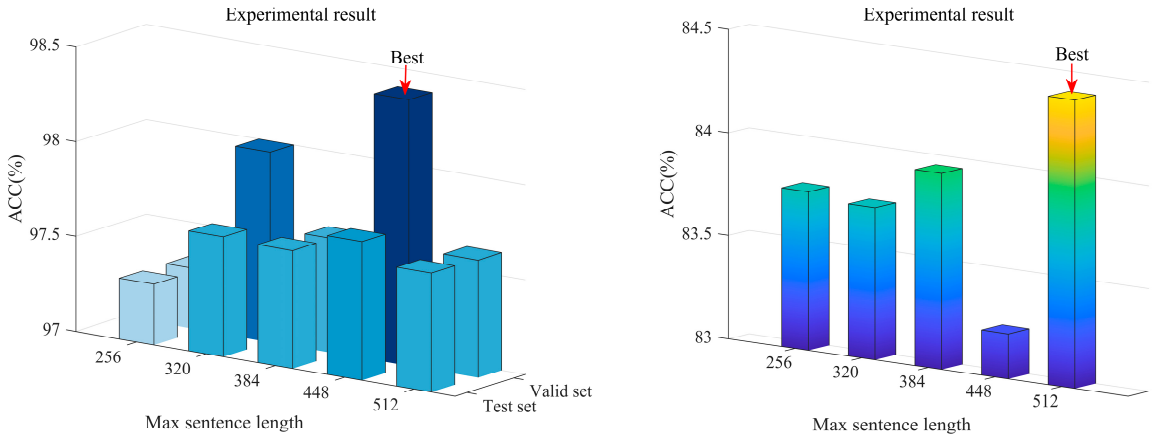



(**a**) (**b**)

**Figure 5.** Ablation study on the activation functions in the feature output layer. (**a**) Study on the THUCNews dataset. (**b**) Study on the arXiv-10 dataset.

(2) **Influence of maximum sequence length on classification:**

We conducted comparative experiments by setting several maximum sequence lengths that have efficiently performed in other text classification tasks. The experimental results are shown in Figure 6.




(**a**) (**b**)

**Figure 6.** Influence of maximum sequence length on classification performance. The accuracy is computed for the test and validation sets on the THUCNews dataset and is computed for test set on the arXiv-10 dataset, ranging from 256 to a maximum length of 512 with an interval of 64 tokens. (**a**) Study on the THUCNews dataset. (**b**) Study on the arXiv-10 dataset.

Our model is more suitable for shorter text extensions (Figure 6a). Considering the structure of the text, merging news headlines and news content into one text frequently results in only half of the information of the news article being fully included in sequences of 256 and 384 tokens, increasing the ambiguity in the model's understanding of headlines. The model's classification ability improves by 0.3% for 384 tokens compared with 256 tokens and by 0.3% for 448 tokens compared with 384 tokens as the maximum sequence length increases. This result demonstrates that longer sequence lengths enable the tokenizer to encode richer textual information, aiding the model in fully learning long-term semantic relationships. However, padding more zero elements at the end of the text introduces noise and interferes with text learning with the increase in the sequence length. The accuracy decreases by 0.1% with the addition of 64 encoding positions, confirming the

above conclusion. The accuracy with a maximum sentence length of 256 is only 0.03% lower compared with a maximum sentence length of 320, while it differs by 0.2% compared with a maximum sentence length of 384 (Figure 6b). The results from the THUCNews dataset demonstrate that gradually increasing the sentence length enables the model to learn richer semantics, building upon the text titles. However, at a sentence length of 448, complete semantic errors in segmentation occurred with further increase in text length. Considering the text structure of arXiv-10, a possible reason is that, at a length of 448 characters, the conclusion part in the abstract is segmented, resulting in incomplete dependence between the title and the full text. This phenomenon results in a decrease in accuracy by 0.7% compared with 320 characters. Finally, extending the sentence length to 512 allows the learning sequence to include the entire text, enabling the model to fully learn the overall semantics and detailed classification clues in the samples. This mechanism yields the best performance among all sentence length strategies, achieving an accuracy of 84.4%.

(3) **Influence of residual connection depth on classification:**

We conducted comparative experiments on two datasets by setting six groups of residual connection depths, with depth increasing by increments of two layers. The experimental results are shown in Figure 7.

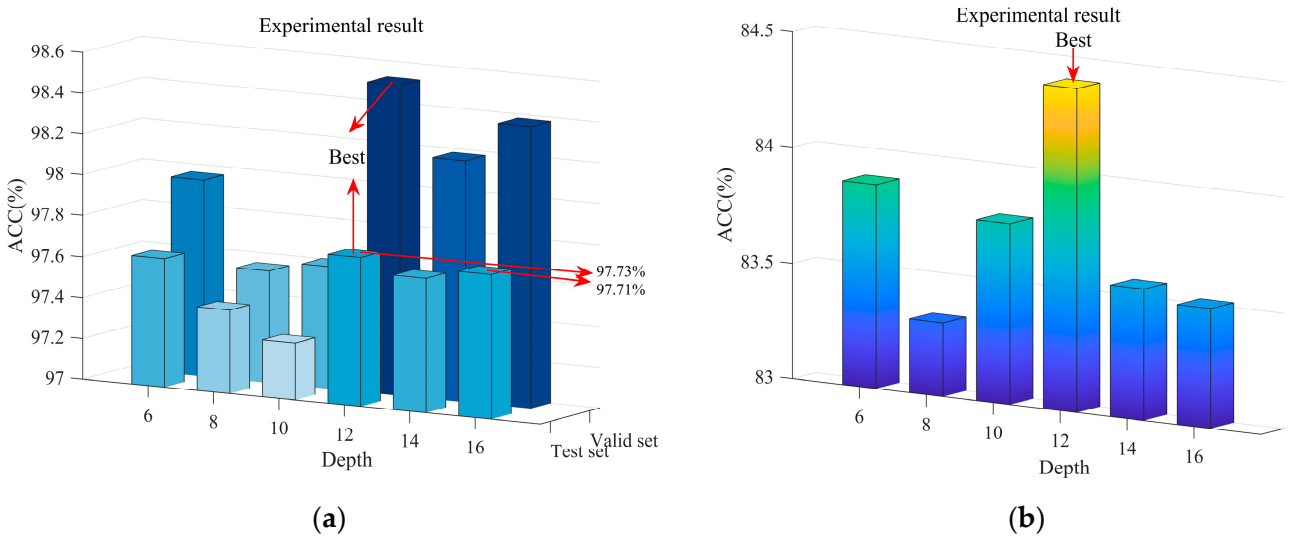



**Figure 7.** Residual connection depth and its influence on classification performance. (**a**) Study on the THUCNews dataset. The performance metrics include accuracy on the test and validation sets (**b**) Study on the arXiv-10 dataset. The performance metrics include accuracy on the test set.

The experimental results indicate that the chosen residual connection depth performs best across samples in two different languages and styles. In Figure 7a, the performance at depths 12 and 16 significantly outperforms other residual connection depths. Moreover, in the comparison between these two depths, the validation accuracy remains consistent between 16 layers and 12 layers. However, the model with 12 layers demonstrates advantages in terms of test set accuracy and F1 score over the larger-parameter model with 16 layers. Additionally, the RQ-OSPTrans with 16 layers occupies 67 GB of GPU memory. Meanwhile, the model with 12 layers requires only 51 GB of GPU memory under the same batch size conditions. Overall, the Transformer encoder with 12 layers of residual connections can sufficiently learn classification clues in repeated questioning and achieve satisfactory performance without the need for further parameter expansion to learn long-range semantics.

In Figure 7b, increasing the depth of residual connections is no longer an improvement method. The RQ-OSPTrans with 16 layers exhibits significant performance degradation compared with the model with 12 layers, indicating that a depth of 12 layers can balance the model parameters and classification performance. Considering all depth selections

across the two datasets, a depth of 12 layers can balance accuracy, model parameter size, F1 score, and other metrics. This condition enables the model to grasp long-range semantic correlations and classification clues in 12 repetitions of questioning, proving that our choice is the optimal strategy.

## 5. Conclusions

The work proposes a text classification method, RQ-OSPTrans, based on the residual-connected Transformer encoder and parallel networks to address the challenge of accurately identifying topics in long texts containing colloquial expressions and noise. The experimental results demonstrate that our method achieves outstanding performance in topic recognition for texts longer than or equal to 256 tokens. Specifically, we achieve validation accuracies of 98.5%, 94.8%, and 84.4% on the THUCNews, AG's News, and arXiv-10 datasets, respectively. These results indicate a significant performance improvement over state-of-the-art pre-trained models in Chinese tasks, while also demonstrating competitiveness in English tasks. However, an important part of RQ-OSPTrans' capabilities comes from pre-trained models, which results in it being more computationally expensive than deep-learning models for other topic classification tasks. In future research, we aim to endow RQ-OSPTrans with multimodal recognition capabilities by adjusting the input-embedding module and improving the accuracy of the domain dataset. One possible approach is to replace the BERT model with the backbone network of the Swin-transformers family, so that RQ-OSPTrans can improve some domain-specific classification problems on image classification tasks, such as the CUB-2011-200 bird fine-grained classification task. This strategy may allow RQ-OSPTrans to be used as an auxiliary model for LLMs, revealing the emergent power that can be brought about by high computational costs. In addition, we will also try to continue to train RQ-OSPTrans on automatic review application data such as social media speech dataset and sentiment analysis dataset, and explore the performance improvement of multiple text classification tasks under the same computing cost, so as to improve the data review efficiency of related industries.

**Author Contributions:** Conceptualization, Y.T., Q.L. and Z.C.; methodology, Y.T. and Z.C.; validation, Y.T. and Q.L.; formal analysis, Y.T., Z.C. and H.L.; investigation, Y.T, Z.C. and H.L.; resources, Y.T. and Z.C.; data curation, Y.T., Q.L., Z.C. and H.L.; writing—original draft preparation, Y.T. and Q.L.; writing—review and editing, Y.T., Q.L., Z.C. and H.L.; supervision, Z.C., T.L. and H.L.; project administration, Y.T. and Z.C.; funding acquisition, Z.C., H.L., T.L. and S.W. All authors have read and agreed to the published version of the manuscript.

**Funding:** This work was supported by the National Natural Science Foundation of China (grant No. 62277026), the Research Project of National Collaborative Innovation Experimental Base for Teacher Development of Central China Normal University (grant No. CCNUTEIII 2021-21), the National Natural Science Foundation of China (grant 62277041, grant 62211530433, grant 62177018), and in part by the National Natural Science Foundation of Hubei Province project (No. 2022CFB529, 2022CFB971), the Jiangxi Provincial Natural Science Foundation under Grant (No. 20232BAB212026), the University Teaching Reform Research Project of Jiangxi Province (Grant No. JXJG-23-27-6), and the Shenzhen Science and Technology Program under Grant (No. JCYJ20230807152900001).

**Institutional Review Board Statement:** Not applicable.

**Informed Consent Statement:** Not applicable.

**Data Availability Statement:** The dataset used in this article are available upon request from the corresponding author (For academic research use only). The data are not publicly available due to privacy.

**Conflicts of Interest:** The authors declare no conflicts of interest.

## References


1. Devlin, J.; Chang, M.-W.; Lee, K.; Toutanova, K. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. In Proceedings of the North American Chapter of the Association for Computational Linguistics, Minneapolis, MN, USA, 2–7 June 2019.
2. Boutell, M.R.; Luo, J.; Shen, X.; Brown, C.M. Learning multi-label scene classification. *Pattern Recognit.* **2004**, *37*, 1757–1771. [CrossRef]
3. Read, J.; Pfahringer, B.; Holmes, G.; Frank, E. Classifier chains for multi-label classification. *Mach. Learn.* **2011**, *85*, 333–359. [CrossRef]
4. Fürnkranz, J.; Hüllermeier, E.; Loza Mencía, E.; Brinker, K. Multilabel classification via calibrated label ranking. *Mach. Learn.* **2008**, *73*, 133–153. [CrossRef]
5. Elisseeff, A.; Weston, J. A kernel method for multi-labelled classification. In Proceedings of the 14th International Conference on Neural Information Processing Systems: Natural and Synthetic, Vancouver, BC, Canada, 3–8 December 2001.
6. Ghamrawi, N.; McCallum, A. Collective multi-label classification. In Proceedings of the 14th ACM International Conference on Information and Knowledge Management, Bremen, Germany, 31 October–5 November 2005.
7. Kim, Y. Convolutional Neural Networks for Sentence Classification. In Proceedings of the Conference on Empirical Methods in Natural Language Processing, Doha, Qatar, 25–29 October 2014.
8. Johnson, R.; Zhang, T. Deep Pyramid Convolutional Neural Networks for Text Categorization. In Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics, Vancouver, BC, Canada, 30 July–4 August 2017.
9. Elman, J.L. Finding Structure in Time. *Cogn. Sci.* **1990**, *14*, 179–211. [CrossRef]
10. Hochreiter, S.; Schmidhuber, J. Long Short-Term Memory. *Neural Comput.* **1997**, *9*, 1735–1780. [CrossRef]
11. Chung, J.; Gulcehre, C.; Cho, K.; Bengio, Y. Empirical evaluation of gated recurrent neural networks on sequence modeling. In Proceedings of the NIPS 2014 Workshop on Deep Learning, Montreal, QC, Canada, 8–13 December 2014.
12. Wang, J.; Yang, Y.; Mao, J.; Huang, Z.; Huang, C.; Xu, W. CNN-RNN: A Unified Framework for Multi-label Image Classification. In Proceedings of the 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, USA, 27–30 June 2016.
13. Yang, Z.; Yang, D.; Dyer, C.; He, X.; Smola, A.; Hovy, E.H. Hierarchical Attention Networks for Document Classification. In Proceedings of the North American Chapter of the Association for Computational Linguistics, San Diego, CA, USA, 12–17 June 2016.
14. You, R.; Zhang, Z.; Wang, Z.; Dai, S.; Mamitsuka, H.; Zhu, S. AttentionXML: Label Tree-based Attention-Aware Deep Model for High-Performance Extreme Multi-Label Text Classification. In Proceedings of the Neural Information Processing Systems, Montréal, QC, Canada, 3–8 December 2018.
15. Zeng, P.; Lin, S.; Sun, H.; Zhou, D. Exploiting Hierarchical Label Information in an Attention-Embedding, Multi-Task, Multi-Grained, Network for Scene Classification of Remote Sensing Imagery. *Appl. Sci.* **2022**, *12*, 8705. [CrossRef]
16. Chen, Z.; Li, J.; Liu, H.; Wang, X.; Wang, H.; Zheng, Q. Learning multi-scale features for speech emotion recognition with connection attention mechanism. *Expert Syst. Appl.* **2023**, *214*, 118943. [CrossRef]
17. Zhou, C.; Sun, C.; Liu, Z.; Lau, F.C.M. A C-LSTM Neural Network for Text Classification. *Comput. Sci.* **2015**, *1*, 39–44.
18. Lai, S.; Xu, L.; Liu, K.; Zhao, J. Recurrent Convolutional Neural Networks for Text Classification. In Proceedings of the AAAI Conference on Artificial Intelligence, Austin, TX, USA, 25–30 January 2015.
19. Zhou, P.; Qi, Z.; Zheng, S.; Xu, J.; Bao, H.; Xu, B. Text Classification Improved by Integrating Bidirectional LSTM with Two-dimensional Max Pooling. In Proceedings of the COLING 2016—The 26th International Conference on Computational Linguistics: Technical Papers, Osaka, Japan, 11–16 December 2016.
20. Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A.N.; Kaiser, Ł.; Polosukhin, I. Attention is all you need. In Proceedings of the 31st International Conference on Neural Information Processing Systems, Long Beach, CA, USA, 4–9 December 2017.
21. Liu, H.; Zhang, C.; Deng, Y.; Liu, T.; Zhang, Z.; Li, Y. Orientation Cues-Aware Facial Relationship Representation for Head Pose Estimation via Transformer. *IEEE Trans. Image Process.* **2023**, *32*, 6289–6302. [CrossRef] [PubMed]
22. Liu, H.; Zhang, C.; Deng, Y.; Xie, B.; Liu, T.; Zhang, Z.; Li, Y. TransIFC: Invariant Cues-aware Feature Concentration Learning for Efficient Fine-grained Bird Image Classification. *IEEE Trans. Multimedia* **2023**. [CrossRef]
23. Yang, Z.; Dai, Z.; Yang, Y.; Carbonell, J.G.; Salakhutdinov, R.; Le, Q.V. XLNet: Generalized Autoregressive Pretraining for Language Understanding. In Proceedings of the Neural Information Processing Systems, Vancouver, BC, Canada, 8–14 December 2019.
24. Liu, Y.; Ott, M.; Goyal, N.; Du, J.; Joshi, M.; Chen, D.; Levy, O.; Lewis, M.; Zettlemoyer, L.; Stoyanov, V. RoBERTa: A Robustly Optimized BERT Pretraining Approach. *arXiv* **2019**, arXiv:1907.11692.
25. Lan, Z.; Chen, M.; Goodman, S.; Gimpel, K.; Sharma, P.; Soricut, R. ALBERT: A Lite BERT for Self-supervised Learning of Language Representations. In Proceedings of the 7th International Conference on Learning Representations, New Orleans, LA, USA, 6–9 May 2019.
26. Cui, Y.; Che, W.; Liu, T.; Qin, B.; Wang, S.; Hu, G. Revisiting Pre-Trained Models for Chinese Natural Language Processing. In Proceedings of the Findings of the Association for Computational Linguistics: EMNLP 2020, Virtual, 16–20 November 2020; pp. 657–668.

27. Kim, S.; Gholami, A.; Yao, Z.; Mahoney, M.W.; Keutzer, K. I-BERT: Integer-only BERT Quantization. In Proceedings of the International Conference on Machine Learning, Virtual, 18–24 July 2021.
28. Reimers, N.; Gurevych, I. Sentence-BERT: Sentence Embeddings using Siamese BERT-Networks. In Proceedings of the Conference on Empirical Methods in Natural Language Processing, Hong Kong, China, 3–7 November 2019.
29. Bai, H.; Zhang, W.; Hou, L.; Shang, L.; Jin, J.; Jiang, X.; Liu, Q.; Lyu, M.R.; King, I. BinaryBERT: Pushing the Limit of BERT Quantization. In Proceedings of the Annual Meeting of the Association for Computational Linguistics, Virtual, 5–10 July 2020.
30. Banino, A.; Badia, A.P.; Walker, J.C.; Scholtes, T.; Mitrovic, J.; Blundell, C. CoBERL: Contrastive BERT for Reinforcement Learning. In Proceedings of the International Conference on Learning Representations, Virtual, 3–7 May 2021.
31. Sanh, V. DistilBERT, a distilled version of BERT: Smaller, faster, cheaper and lighter. In Proceedings of the Thirty-Third Conference on Neural Information Processing Systems (NIPS2019), Vancouver, BC, Canada, 8–14 December 2019.
32. Jiao, X.; Yin, Y.; Shang, L.; Jiang, X.; Chen, X.; Li, L.; Wang, F.; Liu, Q. TinyBERT: Distilling BERT for Natural Language Understanding. In Proceedings of the Empirical Methods in Natural Language Processing, Hong Kong, China, 3–7 November 2019.
33. Joshi, M.; Chen, D.; Liu, Y.; Weld, D.S.; Zettlemoyer, L.; Levy, O. SpanBERT: Improving Pre-training by Representing and Predicting Spans. *Trans. Assoc. Comput. Linguist.* **2019**, *8*, 64–77. [CrossRef]
34. Wang, H.; Ma, S.; Dong, L.; Huang, S.; Zhang, D.; Wei, F. DeepNet: Scaling Transformers to 1000 Layers. *arXiv* **2022**, arXiv:2203.00555.
35. Liu, H.-I.; Chen, W.-L. X-Transformer: A Machine Translation Model Enhanced by the Self-Attention Mechanism. *Appl. Sci.* **2022**, *12*, 4502. [CrossRef]
36. Sun, Y.; Dong, L.; Huang, S.; Ma, S.; Xia, Y.; Xue, J.; Wang, J.; Wei, F. Retentive Network: A Successor to Transformer for Large Language Models. In Proceedings of the Twelfth International Conference on Learning Representations, Vienna, Austria, 7–11 May 2024.
37. Radford, A.; Narasimhan, K.; Salimans, T.; Sutskever, I. Improving Language Understanding by Generative Pre-Training. Available online: https://cdn.openai.com/research-covers/language-unsupervised/language_understanding_paper.pdf (accessed on 22 March 2024).
38. Du, Z.; Qian, Y.; Liu, X.; Ding, M.; Qiu, J.; Yang, Z.; Tang, J. GLM: General Language Model Pretraining with Autoregressive Blank Infilling. In Proceedings of the Annual Meeting of the Association for Computational Linguistics, Bangkok, Thailand, 1–6 August 2021.
39. Pennington, J.; Socher, R.; Manning, C.D. GloVe: Global Vectors for Word Representation. In Proceedings of the Conference on Empirical Methods in Natural Language Processing, Doha, Qatar, 25–29 October 2014.
40. Mikolov, T.; Chen, K.; Corrado, G.S.; Dean, J. Efficient Estimation of Word Representations in Vector Space. In Proceedings of the International Conference on Learning Representations, Scottsdale, AZ, USA, 2–4 May 2013.
41. Dosovitskiy, A.; Beyer, L.; Kolesnikov, A.; Weissenborn, D.; Zhai, X.; Unterthiner, T.; Dehghani, M.; Minderer, M.; Heigold, G.; Gelly, S. An Image is Worth 16 $\times$ 16 Words: Transformers for Image Recognition at Scale. In Proceedings of the International Conference on Learning Representations, Virtual, 3–7 May 2021.
42. Misra, D. Mish: A Self Regularized Non-Monotonic Activation Function. In Proceedings of the 31st British Machine Vision Conference, Virtual, UK, 7–10 September 2020.
43. Li, J.; Sun, M. Scalable Term Selection for Text Categorization. In Proceedings of the Conference on Empirical Methods in Natural Language Processing, Prague, Czech Republic, 28–30 June 2007.
44. Zhang, X.; Zhao, J.J.; LeCun, Y. Character-level Convolutional Networks for Text Classification. In Proceedings of the Neural Information Processing Systems, Montreal, QC, Canada, 7–12 December 2015.
45. Farhangi, A.; Sui, N.; Hua, N.; Bai, H.; Huang, A.; Guo, Z. Protoformer: Embedding prototypes for transformers. In Proceedings of the Pacific-Asia Conference on Knowledge Discovery and Data Mining, Chengdu, China, 16–19 May 2022.
46. Kingma, D.; Ba, L. Adam: A Method for Stochastic Optimization. In Proceedings of the 2nd International Conference on Learning Representations, Banff, AB, Canada, 14–16 April 2014.
47. Hua, J.; Sun, D.; Hu, Y.; Wang, J.; Feng, S.; Wang, Z. Heterogeneous Graph-Convolution-Network-Based Short-Text Classification. *Appl. Sci.* **2024**, *14*, 2279. [CrossRef]
48. Liu, H.; Liu, T.; Chen, Y.; Zhang, Z.; Li, Y. EHPE: Skeleton Cues-based Gaussian Coordinate Encoding for Efficient Human Pose Estimation. *IEEE Trans. Multimed.* **2024**, 124–138. [CrossRef]
49. Liu, T.; Liu, H.; Yang, B.; Zhang, Z. LDCNet: Limb Direction Cues-aware Network for Flexible Human Pose Estimation in Industrial Behavioral Biometrics Systems. *IEEE Trans. Ind. Inf.* **2024**, 1–11. [CrossRef]
50. Ge, B.; He, C.; Xu, H.; Wu, J.; Tang, J. Chinese News Text Classification Method via Key Feature Enhancement. *Appl. Sci.* **2023**, *13*, 5399. [CrossRef]
51. Yue, X.; Zhou, T.; He, L.; Li, Y. Research on Long Text Classification Model Based on Multi-Feature Weighted Fusion. *Appl. Sci.* **2022**, *12*, 6556. [CrossRef]
52. Sun, Y.; Wang, S.; Li, Y.; Feng, S.; Chen, X.; Zhang, H.; Tian, X.; Zhu, D.; Tian, H.; Wu, H. ERNIE: Enhanced Representation through Knowledge Integration. In Proceedings of the Association for Computational Linguistics, Florence, Italy, 28 July–2 August 2019.

53. Clark, K.; Luong, M.-T.; Le, Q.V.; Manning, C.D. Electra: Pre-Training Text Encoders as Discriminators Rather than generators. In Proceedings of the 8th International Conference on Learning Representations, Addis Ababa, Ethiopia, 26–30 April 2020.
54. Radford, A.; Wu, J.; Child, R.; Luan, D.; Amodei, D.; Sutskever, I.; Dean, J.; Ghemawat, S. Language Models are Unsupervised Multitask Learners. In Proceedings of the OSDI'04: Sixth Symposium on Operating System Design and Implementation, Berkeley, CA, USA, 6–8 December 2004.