


Enhancing BERT with Prompt Tuning and Denoising Disentangled Attention for Robust Text Classification

Qingjun Mao ^a

School of Statistics and Mathematics, Zhejiang Gongshang University, Hangzhou, Zhejiang, China

Keywords: Prompt Tuning, BERT, Disentangled Attention, Content-Position Interaction Information.


Abstract: With the increasing complexity of internet data, expanding model parameters and fine-tuning entire models have become inefficient, particularly under limited computational resources. This paper proposes a novel model, Prompt-enhanced Bidirectional Encoder Representation from Transformers (BERT) with Denoising Disentangled Attention Layer (PE-BERT-DDAL), and introduces an efficient fine-tuning strategy to address these challenges. The approach enhances BERT's robustness in handling complex data while reducing computational costs. Specifically, the study introduces a dynamic deep prompt tuning technique within BERT and incorporates a Denoising Disentangled Attention Layer (DDAL) to improve the model's ability to denoise and manage content-position interaction information. The implementation of deep prompt tuning facilitates the model's rapid adaptation to downstream tasks, while DDAL strengthens content comprehension. Comparative experiments are conducted using three datasets: Twitter entity sentiment analysis, fake news detection, and email spam detection. The results demonstrate that PE-BERT-DDAL outperforms baseline models in terms of accuracy and loss reduction, achieving an average peak accuracy of 0.9059. PE-BERT-DDAL also improves accuracy by 6.75%, 5.93%, and 8.97%, respectively, over baseline models. These findings validate PE-BERT-DDAL's effectiveness, showcasing its capacity for rapid task adaptation and robustness in complex data environments.

1 INTRODUCTION

Natural language processing (NLP) is a key research direction in artificial intelligence. It is an interdisciplinary field that interacts between computer science and linguistics. In recent years, with the explosive growth of text data on social media and the rising demand for conversational artificial intelligence (AI), the NLP field has developed rapidly. In 2017, Vaswani et al. put forward a novel neural network architecture, which was named Transformer (Vaswani, 2017). The Transformer has promptly become a core technology in the NLP field, offering a unified framework for the design and implementation of models across various tasks.

The introduction of the self-attention mechanism marks a remarkable innovation in the Transformer, enabling the model to evaluate the importance of different tokens within a sequence. This mechanism has progressed into the multi-head self-attention variant, which greatly enhances the model's

efficiency in parallel processing and ability to learn diverse features. The Transformer first projects the input queries, keys, and values into different subspaces through linear projections. During this process, these projected queries, keys, and values are fed into attention pooling in parallel, where each of the attention pooling outputs is referred to as a head. The multi-head self-attention mechanism empowers the Transformer to effectively extract features from various dimensions of the sequence. Subsequently, language pre-training models such as Generative Pre-trained Transformer (GPT) and Bidirectional Encoder Representations from Transformers (BERT) were proposed based on the Transformer. Here, this paper briefly explains the working mechanism of BERT, as all subsequent models mentioned in this research are developed from BERT. BERT was proposed by Devlin et al. in 2018 (Devlin, 2018), and its main structure is constructed by stacking the Transformer Encoder parts. In the pre-training phase, BERT has the following tasks: Masked Language

^a <https://orcid.org/0009-0003-7055-0075>

Modeling (MLM) and Next Sentence Prediction (NSP). Leveraging large-scale natural language datasets, BERT can learn rich semantic relations between contexts from these two tasks. BERT exhibits excellent performance in domains such as natural language inference, named entity recognition, and machine translation. Consequently, fine-tuning BERT for downstream tasks has emerged as a popular method.

To further enhance the performance of pre-trained language models in various NLP tasks, researchers have proposed diverse methods based on the BERT architecture. Robustly Optimized BERT Pretraining Approach (RoBERTa) optimizes the training strategy of BERT by removing the NSP task (Liu, 2019). To overcome the limitations of BERT in capturing global bidirectional context, XLNet introduces a permutation language modeling with a bidirectional autoregressive framework (Yang, 2019). ALBERT improves the model performance by implementing parameter decomposition and cross-layer parameter sharing. Additionally, ALBERT replaces the NSP task with Sentence Order Prediction (SOP), considerably enhancing the model's ability to capture sentence coherence (Lan, 2019). ELECTRA no longer uses the [MASK] token for training. Instead, it performs Replaced Token Detection (RTD) tasks and trains a discriminator to predict whether each token in the corrupted input has been replaced by a generator sample (Clark, 2020). Decoding-enhanced-BERT-with-disentangled attention (DeBERTa) incorporates a disentangled attention mechanism and an enhanced masked decoder, further refining the model's understanding of content and positional information (He, 2020). All of the aforementioned models aim to improve performance by modifying the operational mechanisms of BERT. Furthermore, researchers have carried out numerous lightweight enhancements to BERT. Stacked DeBERT adds a Denoising Transformer Layer on top of the BERT architecture, improving the model's robustness to incomplete data (Sergio et. al., 2021). Adapter tuning introduces two Adapter modules into each Transformer layer, allowing the model to be significantly more adaptable to downstream tasks by simply updating only the parameters within the Adapter modules. This approach greatly reduces the cost of fine-tuning (Houlsby et. al., 2019). Similarly, P-tuning introduces learnable prompt tokens into the input, resulting in cost savings and improved efficiency (Liu et. al., 2021).

The focus of this research is on enhancing BERT's performance by integrating advanced attention mechanisms with efficient fine-tuning strategies. The

proposed architecture, Prompt-enhanced BERT with Denoising Disentangled Attention Layer (PE-BERT-DDAL), introduces two key innovations. First, deep prompt tuning is incorporated within the BERT layers based on P-tuning v2, where learnable prompt tokens are prefixed to the input sequence. This deep prompt tuning allows for a more nuanced adaptation of the model across multiple layers, leading to improved context understanding. Second, the Disentangled Attention mechanism from DeBERTa is integrated into the model as part of a new Denoising Transformer Layer, referred to as the DDAL. This layer addresses the positioning bias and content distortion that may arise from prompt tuning, as well as noise from incomplete data. By enhancing the robustness of BERT's output, DDAL refines the model's capacity to handle both noisy and clean data inputs. Experimental results show that fine-tuning only the final layer of BERT yields competitive performance, significantly reducing computational costs. Unlike DeBERTa, which applies disentangled attention across all layers, PE-BERT-DDAL requires fewer parameters, making it a more resource-efficient alternative while maintaining strong classification capabilities.

2 METHODOLOGIES

2.1 Dataset Description and Preprocessing

This study utilizes three Kaggle datasets for different classification tasks: sentiment classification, fake news detection, and spam detection. The Twitter Entity Sentiment Analysis Dataset includes tweets with sentiment labels (positive, negative, or neutral). Tweets are often unstructured and contain noise such as emojis, spelling errors, and slang, which renders this dataset appropriate for testing the model's capability to handle complex and noisy text. The Fake News Detection Dataset consists of textual content labeled as real or fake, where fake news is characterized by ambiguous wording and misleading information. The model is required to comprehend semantics and identify deceptive reasoning in this task. The Email Spam Detection Dataset contains email bodies labeled as spam or non-spam, testing the model's robustness in handling distracting features and promotional language common in spam emails (jp797498e, 2024; iamrahulthorat, 2024; nitishabharathi, 2024).

During data preprocessing, Hyper Text Markup Language (HTML) tags, special characters, and

illegal symbols were removed from the text. The data was then tokenized and transformed into word embedding vectors for BERT input. To maintain consistency, short texts were padded and long texts were truncated to unify sequence lengths.

2.2 Proposed Model

To enhance BERT's performance in text classification tasks, this paper introduces the PE-BERT-DDAL model, as illustrated in Figure 1. The core innovation lies in incorporating a deep Prompt Tuning mechanism within BERT to dynamically inject prompt information across different layers. This approach allows the model to better adapt to diverse downstream tasks by integrating task-specific information directly into the model's architecture. Additionally, the Disentangled Attention Mechanism, inspired by the DeBERTa model, is applied to improve the denoising Transformer layer, leading to the development of a novel DDAL. The DDAL is designed to enhance the model's robustness by correcting content distortions and addressing position biases introduced during prompt tuning, especially when handling incomplete or noisy data. In the classification tasks, PE-BERT-DDAL employs a lightweight single-layer fine-tuning strategy, which optimizes only the final layer of the model. This reduces the need for extensive computational resources while maintaining strong classification performance, making it both efficient and effective for real-world applications.

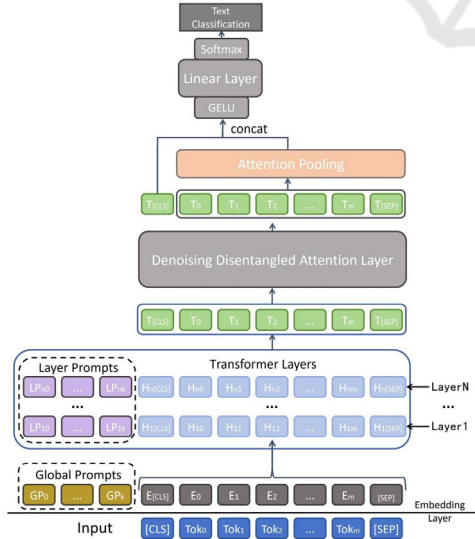


Figure 1: The architecture and workflow of the PE-BERT-DDAL model (Picture credit: Original).

This research employs the conventional embedding layer of BERT to convert each token in the input text into a word embedding vector applicable to the BERT model. Specifically, the embeddings of the BERT input sequence consist of the sum of token embeddings, segment embeddings, and positional embeddings. Token embeddings encapsulate the semantic characteristics of each token, segment embeddings serve to distinguish between sentences, while positional embeddings encode the positional information of tokens within the sequence. The input text tokens in this paper are represented as follows:

$$\{[CLS], Tok_0, \dots, Tok_m, [SEP]\} \quad (1)$$

where Tok_i ($i = 1, 2, \dots, m$) represents each token in the text, and m is the maximum length of the input sequence. Each Tok_i is transformed into an embedding vector after passing through the embedding layer:

$$E_i = E_{token}(Tok_i) + E_{segment}(Tok_i) + E_{position}(Tok_i) \quad (2)$$

$E_{token}(Tok_i)$ represents the token embedding, $E_{segment}(Tok_i)$ represents the segment embedding of the token, and $E_{position}(Tok_i)$ represents the positional embedding of the token. The embeddings of the input text sequence are represented as follows:

$$\{E_{[CLS]}, E_0, \dots, E_m, E_{[SEP]}\} \quad (3)$$

The model employs conventional multi-layer bidirectional transformers to extract and learn contextual semantics from the embedding vectors of the input text. This study introduces the deep prompt tuning technique while freezing most of the layers. Specifically, after the input sequence is converted into embedding vectors, this study initially adds learnable prefix prompts at the front, namely global prompts. In each layer of the Transformer, the input embedding sequence is generated into the hidden state vector through the self-attention mechanism and feed-forward neural network. In this paper, the output hidden state vectors of a given layer are represented as follows:

$$h_j = \{H_{j[CLS]}, H_{j0}, \dots, H_{jm}, H_{j[SEP]}\} \quad (4)$$

where H_{ji} represents the hidden state of the i th token in the j th layer, with n being the number of Transformer layers. $H_{j[CLS]}$ and $H_{j[SEP]}$ represent

the hidden states of the [CLS] token and the [SEP] token, respectively. This study also adds learnable prefix prompts to the hidden state before each layer, forming a new hidden state h'_j , namely layer prompts. Then, h'_j is fed into the next Transformer layer to further learn contextual features. The hidden states of the input sequence after being processed by multiple Transformer layers are represented as follows:

$$\{T_{[CLS]}, T_0, \dots, T_m, T_{[SEP]}\} \quad (5)$$

The sequence is then input into the DDAL to eliminate noise, correct positioning information bias, and further strengthen the connection between content and positioning information. A detailed explanation of the specific operational mechanism will be provided in the subsequent sections.

After DDAL processes the sequence, the content information at different positions has varying degrees of importance. Even though the conventional BERT model uses the [CLS] token as a feature vector for classification tasks, this paper argues that relying solely on the information in the [CLS] token is limited. Therefore, this study adopts attention pooling to perform a weighted average on the entire sequence and calculate a global feature vector that represents the entire sequence, thus compensating for the information that the [CLS] token fails to capture. The formula of attention pooling is shown as follows:

$$v = \sum_{i=1}^m \alpha_i T_i \quad (6)$$

where v is the final global feature representation, T_i is the hidden state of the i th token in the sequence, and α_i represents the attention weight of the token.

Ultimately, $T_{[CLS]}$ and v are concatenated along the feature dimension. The resulting concatenated feature vector is subsequently processed through the GELU activation function, followed by a linear layer and a SoftMax layer to yield the final output.

2.2.1 Deep Prompt Tuning

This study utilizes the deep prompt tuning technique to minimize the number of parameters required for fine-tuning, thereby reducing computational resource expenditures. This approach facilitates the efficient fine-tuning of BERT models for downstream tasks, even under constrained computational resources. The P-tuning v2 findings indicate that deep prompt tuning can achieve the performance of conventional fine-tuning by adjusting only 0.1% to 3% of the parameters in billion-parameter models (Liu et. al., 2021). Notably, deep prompt tuning can guide pre-

trained models to extract features from sequences through the leverage of learnable prompts, without greatly altering the original parameters. Given that the embedding vectors corresponding to the input sequence are as follows:

$$\{E_{[CLS]}, E_0, \dots, E_m, E_{[SEP]}\} \quad (7)$$

In this study, global prompts are added at the beginning of the embedding sequence, forming a new sequence X' :

$$X' = \{GP_1, \dots, GP_k, E_{[CLS]}, E_0, \dots, E_m, E_{[SEP]}\} \quad (8)$$

where GP_i represents the i th global prompt information, and k represents the length of the prompts. Afterward, this new sequence is fed into the multi-layer Transformer for contextual feature extraction. The layer prompts introduced in each layer of the Transformer are specifically represented as follows:

$$h'_j = \{LP_{j0}, \dots, LP_{jk}, H_{[CLS]}, H_{j0}, \dots, H_{jm}, H_{j[SEP]}\} \quad (9)$$

These prompts enhance the model's dynamic adjustment of feature representations across multiple layers, thereby improving the model's adaptability.

2.2.2 DDAL

This paper integrates the denoising Transformer layer with a disentangled attention mechanism to propose DDAL, with the objective of augmenting the model's capability to manage noise and address incomplete data. Meanwhile, the integration of a disentangled attention mechanism significantly bolsters the model's proficiency in handling and interpreting positional information. Specifically, the disentangled attention mechanism can redress the positional information bias brought about by deep prompt tuning and mitigate the loss of positional information in sequences after the sequence is reconstructed by the denoising Transformer layer. Furthermore, it can enhance the Transformer's comprehension of the interaction between content and position information, thereby mitigating biases arising from the coupling between content and position information. The architecture of DDAL is shown in the Figure 2.

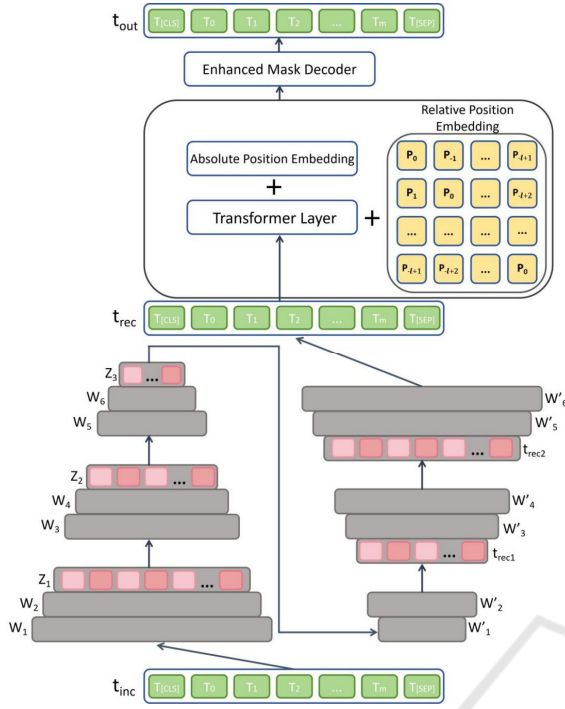


Figure 2: DDAL architecture (Picture credit: Original).

In DDAL, the input sequence is initially processed by a two-stack three-layer MLP, which compresses and reconstructs the sequence information. Specifically, given an incomplete embedding representation with noise, t_{inc} , the MLP will first compress it into a low-dimensional representation z , and then reconstruct it into a complete embedding representation t_{rec} . Here, W_1, W_2, \dots, W_6 are weight matrices, b_1, b_2, \dots, b_6 are bias terms. Similarly, W'_1, W'_2, \dots, W'_6 are weight matrices, b'_1, b'_2, \dots, b'_6 are bias terms.

Building upon the reconstructed embedding vectors, this study introduces the disentangled attention mechanism to individually handle content and positional information. Prior to a formal explanation of the disentangled attention mechanism, it is essential to first introduce the concept of relative position embedding. Suppose any two tokens in the given sequence, denoted as $token_i$ and $token_j$, their relative distance $\omega(i,j)$ is obtained by the following formula:

$$\omega(i,j) = \begin{cases} 0 & \text{for } i-j \leq -l \\ 2l-1 & \text{for } i-j \geq l \\ i-j+1 & \text{others} \end{cases} \quad (10)$$

where l represents the maximum relative distance, which is a hyperparameter.

Subsequently, the relative position embeddings P_i and P_j can be generated by the relative distance $\omega(i,j)$, typically utilizing a learnable embedding layer, which is adapted by the model according to the specific task. The details are shown as follows:

$$P_{\omega(i,j)} = \text{Embedding}(\omega(i,j)) \quad (11)$$

Relative position embedding enhances the model's ability to capture local features within the sequence. This paper will explain the operation of the disentangled attention mechanism grounded in relative position embedding. The calculation formula for the disentangled attention mechanism is shown as follows:

$$A_{i,j} = t_{rec_i} t_{rec_j}^T + t_{rec_i} P_j^T + P_i t_{rec_j}^T \quad (12)$$

t_{rec_i} and t_{rec_j} represent the reconstructed embeddings of $token_i$ and $token_j$, respectively. $t_{rec_i} t_{rec_j}^T$ represents the content interaction between two tokens. $t_{rec_i} P_j^T$ represents the content-to-position interaction, and $P_i t_{rec_j}^T$ represents the position-to-content interaction. After the relative position embedding, the absolute position embedding is continuously added to the embedding vectors of the sequence. Finally, the sequence passes through the enhanced mask decoder to generate the output t_{out} .

Learnable prompt embeddings can guide models to achieve higher performance, without having natural language semantics like input sequences, resulting in a certain degree of semantic confusion and information bias. In addition, the dispersed attention mechanism effectively solves the problem of positional information loss within the sequence during the compression and reconstruction process of MLP by independently processing content information and positional data.

2.2.3 Loss Function

The main objective of this study is text classification, which involves the prediction of category labels for given textual data. The cross-entropy loss function provides an efficient means to assess how closely the model's predicted probability distribution aligns with the true labels. Given that this study involves a multi-class classification task, the cross-entropy loss function is a highly suitable choice. It is defined as follows:

$$L_{class} = -\sum_{i=1}^V y_i \log \hat{y}_i \quad (13)$$

where V represents the number of samples, y_i is the true label of the i th sample, \hat{y}_i is the predicted label of the i th sample.

To avoid the situation where the model assigns unequal weights to content information and location information, this study introduces a regularization term to balance the attention weights of the interaction information in the disentangled attention mechanism. The loss function can be further improved in the following form:

$$L = -\sum_{i=1}^V y_i \log \hat{y}_i + \lambda \sum_{i,j} \|A_{i,j}\| \quad (14)$$

where λ is the regularization coefficient, and $A_{i,j}$ is the weight matrix in the disentangled attention mechanism. The introduction of this regularization term ensures that the model will not excessively favor any particular interaction during training, thus avoiding overfitting and further enhancing the model's generalization ability.

2.3 Implementation Details

BERT mainly has two application approaches: fine-tuning and feature extraction. Full fine-tuning adapts well to downstream tasks but is highly resource-intensive. On the other hand, using BERT solely as a fixed encoder for feature extraction may not yield optimal results. To strike a balance between the two, this study implements a fine-tuning strategy that involves unfreezing part of the encoder layers. This study only unfreezes the last Transformer layer of BERT, while keeping the other layers frozen. The total number of trainable parameters in the model is approximately 14.7 million. In addition, this paper selects three baseline models: BERT_{base}, DeBERTa_{base}, RoBERTa_{base}. To ensure the fairness of the comparative experiments as much as possible, it is necessary for the scale of the learnable parameters in the baseline models to be similar to that of PE-BERT-DDAL. Therefore, The BERT_{base} unfreezes

the last two layers, with 14.1 million trainable parameters. DeBERTa_{base} unfreezes the last two layers and has 16.6 million trainable parameters. RoBERTa_{base} unfreezes the last two layers and has 14.1 million trainable parameters.

The optimization function used in this study is AdamW. For the unfrozen last Transformer layer, the learning rate is set to 1e-5. The learning rate of the deep prompt tuning module is set to 1e-3, and the learning rate of the compression and reconstruction MLP is set to 1e-4.

3 RESULT AND DISCUSSION

In this section, this study conducts a comparative analysis of the performance of the proposed PE-BERT-DDAL model against three baseline models: BERT, DeBERTa, and RoBERTa. Specifically, this study trains the four models on the three datasets of tweet sentiment analysis, fake news detection, and spam classification with a fixed number of epochs, and records the test set accuracy and loss function values of each model during the training process. Subsequently, this study will deeply analyze the performance of each model from perspectives including the training process and the highest performance of models. It is worth noting that the following figures from left to right correspond to Twitter sentiment analysis, fake news detection, and spam classification.

3.1 Analysis of Accuracy Variations During Training

As shown in Figure 3, after 10 epochs of training, the test accuracy of PE-BERT-DDAL significantly outperforms the other three baseline models. In the early stages of training, the test set accuracy of PE-BERT-DDAL exhibits a rapid increase and then stabilizes, indicating that PE-BERT-DDAL demonstrates strong adaptability in text classification tasks.

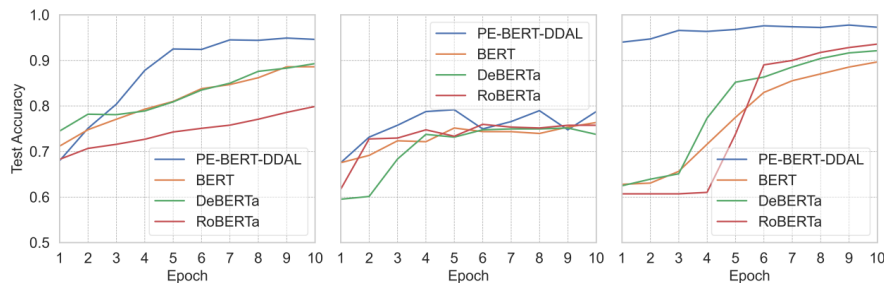


Figure 3: Accuracy variations (Picture credit: Original).

In the Twitter sentiment analysis task, PE-BERT-DDAL begins to converge at the 7th epoch, reaching a test accuracy of approximately 0.95. The BERT and DeBERTa models exhibit similar performance on this task, with their test accuracy approaching 0.90 by the end of the training. In contrast, RoBERTa performed worse, with a maximum test accuracy of only 0.80. In the fake news detection task, the test set accuracy of PE-BERT-DDAL is generally higher than that of the other three models throughout the training process. The test accuracy of PE-BERT-DDAL rapidly increases in the early stages of training, nearly reaching peak accuracy by the 4th epoch. Afterward, its test accuracy begins to fluctuate. The test set accuracy of the DeBERTa and RoBERTa models rises at a relatively slow pace in the early stages of training, while the test set accuracy of the BERT model increases more slowly. The baseline models only reach their optimal accuracy after the 6th epoch. In the spam detection task, the PE-BERT-DDAL model achieves a very high accuracy early in the training and significantly outperforms the other three models. Its test accuracy then gradually improves. However, the other three models show no significant upward trend in the early stages of training, only beginning to rise rapidly from the 4th epoch. After the 6th epoch, the rate of increase in test set accuracy slows down.

3.2 Analysis of Loss Variations During Training

Figure 4 illustrates the loss variations of each model during the training process. The loss reduction of PE-BERT-DDAL is consistently faster than that of the baseline models.

In the Twitter sentiment analysis task, the loss value of PE-BERT-DDAL decreases rapidly. By the end of the 10th epoch of training, its loss value was below 0.2, while the loss values of the other models

were all above 0.5. In the fake news detection task, the loss reduction trends of all four models are basically consistent. However, the loss value of PE-BERT-DDAL is lower than that of the other three models.

In the spam detection task, the loss of PE-BERT-DDAL starts at a low level in the early stages of training and gradually decreases toward zero. The other three models gradually decrease throughout the training process, with DeBERTa and RoBERTa's loss approaching 0.2 at the end of the training, while BERT's loss is between 0.3 and 0.4. The low loss values and high-test accuracy of PE-BERT-DDAL indicate that its dynamic deep prompt tuning and disentangled attention mechanism allow it to not only converge more quickly in noisy and complex data environments but also learn more effective features from the datasets.

3.3 Analysis of Peak Accuracy

To display the generalization ability of each model, this study computes the average highest accuracy across different datasets (see in Table 1). PE-BERT-DDAL performs the best across all tasks with an average accuracy of 0.9059. The average accuracy of the BERT model amounts to 0.8486, that of DeBERTa is 0.8551, and that of RoBERTa is 0.8313. Using the accuracy of the baseline models as a reference, PE-BERT-DDAL improves performance by approximately 6.75%, 5.93%, and 8.97% compared to BERT, DeBERTa, and RoBERTa, respectively.

The experimental results show that PE-BERT-DDAL outperforms the baseline models. During the training, whether the swift increase in test set accuracy or the rapid decrease in loss indicates that deep prompt tuning can guide the model to quickly adapt to downstream tasks. The highest average accuracy indicates that DDAL effectively enhances

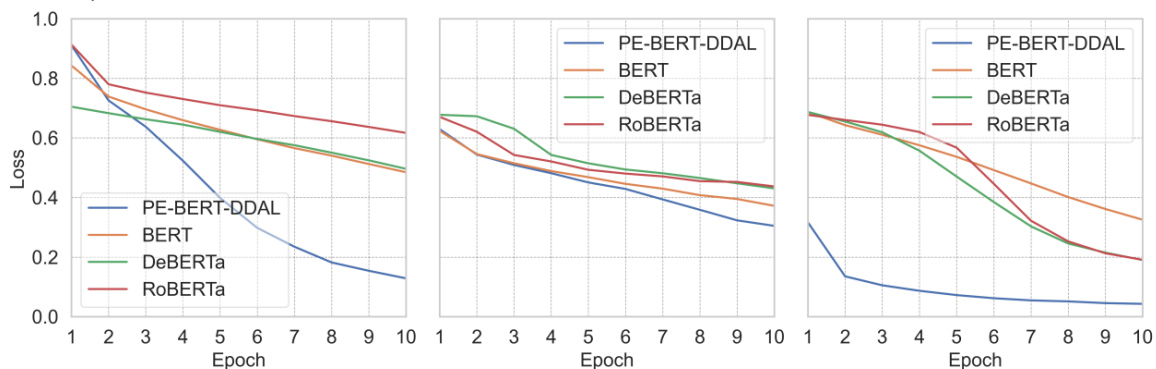


Figure 4: Loss variations (Picture credit: Original).

Table 1: Summary of peak accuracy across datasets.

Model	Tweet (%)	Fake news (%)	Spam (%)	Average Accuracy (%)
PE-BERT-DDAL	0.9489	0.7915	0.9774	0.9059
BERT	0.8858	0.7635	0.8965	0.8486
DeBERTa	0.8928	0.7515	0.9211	0.8551
RoBERTa	0.7987	0.7595	0.9356	0.8313

the model's robustness and competence to capture valuable features. Based on these experimental outcomes, the prediction capability and training efficiency of PA-BERT-DDAL in classification tasks have been thoroughly validated.

4 CONCLUSIONS

This paper presents PE-BERT-DDAL, a novel BERT-based model designed to improve performance under limited computational resources and complex data environments. By integrating deep prompt tuning and the DDAL, the model enhances adaptability to downstream tasks and robustness in processing complex data. Global prompts are added at the input sequence's start, with layer-specific prompts introduced in each Transformer layer. The model then passes through a denoising Transformer layer equipped with a disentangled attention mechanism. Comparative experiments using BERT, DeBERTa, and RoBERTa as baselines were conducted on three datasets: Twitter Entity Sentiment Analysis, Fake News Detection, and Email Spam Detection. Results show that PE-BERT-DDAL outperforms baseline models in accuracy and loss reduction, achieving an average peak accuracy of 0.9059 across datasets. The dynamic deep prompt tuning contributes to faster convergence in early training. This research highlights the model's strong robustness and generalization capabilities. Future work will focus on expanding the model's application to more complex NLP tasks, such as natural language inference and text generation, and exploring advanced fine-tuning techniques and attention mechanisms to further improve its performance.

REFERENCES

- Clark, K. 2020. Electra: Pre-training text encoders as discriminators rather than generators. arXiv preprint arXiv:2003.10555.
- Devlin, J. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. arXiv preprint arXiv:1810.04805.
- He, P., Liu, X., Gao, J., & Chen, W. 2020. Deberta: Decoding-enhanced bert with disentangled attention. arXiv preprint arXiv:2006.03654.
- Houlsby, N., Giurgiu, A., Jastrzebski, S., Morrone, B., De Laroussilhe, Q., Gesmundo, A., ... & Gelly, S. 2019. Parameter-efficient transfer learning for NLP. In International conference on machine learning (pp. 2790-2799). PMLR.
- iamrahulthorath. 2024. fakenews-csv. Kaggle. Retrieved on 2024, Retrieved from: <https://www.kaggle.com/datasets/iamrahulthorath/fakenews-csv>.
- jp797498e. 2024. Twitter-entity-sentiment-analysis. Kaggle. Retrieved on 2024, Retrieved from: <https://www.kaggle.com/datasets/jp797498e/twitter-entity-sentiment-analysis>.
- Lan, Z. 2019. Albert: A lite bert for self-supervised learning of language representations. arXiv preprint arXiv:1909.11942.
- Liu, X., Ji, K., Fu, Y., Tam, W. L., Du, Z., Yang, Z., & Tang, J. 2021. P-tuning v2: Prompt tuning can be comparable to fine-tuning universally across scales and tasks. arXiv preprint arXiv:2110.07602.
- Liu, Y. 2019. Roberta: A robustly optimized bert pretraining approach. arXiv preprint arXiv:1907.11692.
- nitishabharathi. 2024. email-spam-dataset. Kaggle. Retrieved on 2024, Retrieved from: <https://www.kaggle.com/datasets/nitishabharathi/email-spam-dataset>.
- Sergio, G. C., & Lee, M. 2021. Stacked DeBERT: All attention in incomplete data for text classification. Neural Networks, 136, 87-96.
- Vaswani, A. 2017. Attention is all you need. Advances in Neural Information Processing Systems.
- Yang, Z. 2019. XLNet: Generalized Autoregressive Pretraining for Language Understanding. arXiv preprint arXiv:1906.08237.