蒋安祥, 张洁. 融合提示学习和注意力模块的预训练模型研究[J]. 智能计算机与应用,2026,16(2):1-7. DOI:10.20169/j.issn.2095-2163.24040901

融合提示学习和注意力模块的预训练模型研究

蒋安祥, 张 洁
(南京邮电大学 计算机学院、软件学院、网络空间安全学院, 南京 210023)

摘 要: 针对现有模型对文本预测精度不高、小样本训练结果较差等问题,提出一种改进的语言模型 BPFP。首先,在模型嵌入层融合字向量信息和位置编码信息,更好地整合位置信息和语义信息。其次,减少了 Transformer 编码器的数量,提高了训练速度。最后,融合了提示学习模块,提升总体精度的同时在小样本数据集上也取得了不错的效果。实验结果显示,在长文本数据集上,BPFP 模型在遮挡语言模型和下一句预测准确率上分别较主流模型提升 3.7%至 6.9%,在短文本数据集上,准确率提升 5.9%至 12.3%,训练时间缩短了约三分之一。本文研发模型展现出了更强的鲁棒性。

关键词: 自然语言处理; 文本预测; 提示学习; 预训练模型
中图分类号: TP391.1 **文献标志码**: A **文章编号**: 2095-2163(2026)02-0001-07

Research on pretraining model integrating hint learning and attention module

JIANG Anxiang, ZHANG Jie
(School of Computer Science, School of Software, School of Cyberspace Security, Nanjing University of Posts and Telecommunications, Nanjing 210023, China)

Abstract: Aiming at the problems of low text prediction accuracy and poor training results of small sample, an improved language model BPFP is proposed. Firstly, word vector information and position coding information are fused in the model embedding layer to better integrate position information and semantic information. Secondly, the number of Transformer encoders is reduced and the training speed is improved. Finally, the prompt learning module is integrated to improve the overall accuracy and achieve good results on small sample data sets. The experimental results show that the accuracy of BPFP model in occlusion language model and next sentence prediction is improved by 3.7% to 6.9% compared with the mainstream model on the long text data set, and the accuracy is improved by 5.9% to 12.3% on the short text data set, the training time is shortened by about one-third and the model shows stronger robustness.

Key words: natural language processing; text prediction; prompt learning; pretraining models

0 引 言

近十年来,自然语言处理技术经历了一个快速发展过程。在引入神经网络之前文本预测常见的方法包括贝叶斯算法[1]、Viterbi 算法[2]、隐马尔可夫模型等[3]。

随着技术的发展,NLP 领域引进了神经网络,这类方法不用手动设置特征和规则,代表算法有 CNN[4]、RNN[5]、机器翻译中的 Seq2Seq 模型[6]等等,但这类算法需要人工设计合适的神经网络架构来对数据集进行训练。因此引入了预训练模型。Mutinda 等学者[7]提出了一种结合情感词典、N-gram、BERT 和 CNN 的情感分类模型。对于长文本信息的处理,Chen 等学者[8]同样采用了微调模型的方式,提出了一种基于 BERT 的局部特征卷积网络模型,用于预测新闻文本的类别,并已取得了显著成效。但微调模型只是针对模型的参数等进行调整,大型模型的参数发展到现在数量级是极其庞大的,微调参数需要做大量的实验。同时,针对一些小样本或者零样本的数据集,预训练模型的效果也并不明显。

因此相关学者通过合适的 prompt 模板来直接在预训练模型上解决下游任务,这种模式需要极少

基金项目: 国家重点研发计划(2018YFB1500902)。
作者简介: 蒋安祥(2000—),男,硕士研究生,主要研究方向:自然语言处理,文本预测。
通信作者: 张 洁(1981—),女,博士,高级工程师,硕士生导师,主要研究方向:人工智能,自然语言处理。Email:zhangjie@njupt.edu.cn。
收稿日期: 2024-04-09

量下游任务数据,使得小样本、零样本学习成为可能。Yang 等学者[9]针对小样本训练结果不佳的问题提出了一种基于伪标签融合聚类算法的小样本文本分类模型,提升了模型精度,但聚类算法的训练时间过长的问题却没有得到有效解决。Ma 等学者[10]在处理小样本故障诊断的问题时提出了一种基于多变量正态分布和马氏距离的数据生成模型,该模型虽然有很好的生成效果,但是也限制了模型的生成能力。Liu 等学者[11]对于即时调整正常大小的预训练模型表现不佳的问题,采用提示学习的方法已经取得了非常不错的效果,但模型的体积依旧非常庞大,因此仍有进一步提升的空间。

为此,本文提出了一种融合提示学习的微调模型 BPFP(Bert-Prompt-Fine-tuning-Predict),主要工作和创新如下:

(1)首先,BPFP 模型在嵌入层融合了位置信息和字嵌入信息,更好地整合位置信息和语义信息。

(2)其次,减少 Transformer 编码器堆叠数量,在略微降低精度的情况下缩短训练时间。

(3)最后,融合了提示学习模块,用预训练任务的 prompts 去初始化下游任务的 prompts,在提升精度的同时缩短训练时间。

1 BPFP 模型结构

BPFP 模型分为 4 个主要模块:嵌入层、Transformer 编码层、Bert 层以及提示学习层。BFPF 模型结构见图 1。数据经过提示学习模板得到标准化的输出,将该输出输入到预训练模型中,调整模型参数以使其适应新任务。

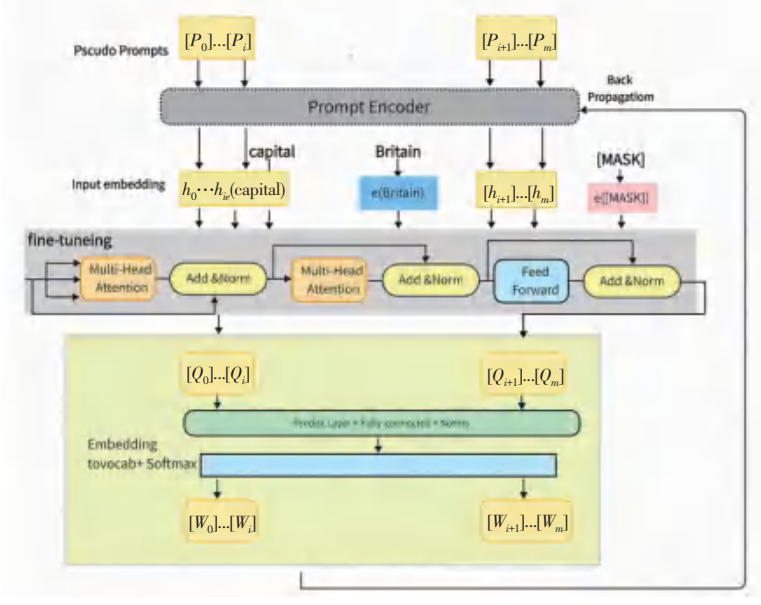

图 1 BPFP 模型结构图

Fig. 1 Diagram of BPFP model structure

1.1 嵌入层

本研究中,构建的模型包含 3 种嵌入层[12]:字符信息嵌入、位置向量和文本向量。这些向量在嵌入空间内相互作用,以生成编码层的输出。

字符信息嵌入针对序列中的每个位置,模型分别计算上述 3 种嵌入信息。由于这 3 种信息在维度上是一致的,初始的词嵌入可以通过这 3 种信息对应元素的相加获得。需要指出的是,在大多数模型中,这 3 种嵌入的组合并不局限于简单的相加,也可以通过逐元素相乘或者使用 Concat 函数来组合。在本文的模型中,采用的是直接相加的方式。相较于其他方法,加法操作在计算上更为简单,并且相对于拼接操作,还减少了计算量。

此外,模型输入除了字向量,还包含文本向量和位置向量,分别用于刻画文本的全局语义信息以及对不同位置的字/词分别附加一个不同的向量来进行区分。

1.2 Transformer 编码层

Attention 机制[13]是 Transformer 中的关键部分。本文的 BPFP 模型也利用了 Attention 机制来构建 Transformer 模块。在此基础上,用多层 Transformer 组装 BPFP 模型。

Attention 机制涉及 3 个核心概念：Query、Key 和 Value。在此背景下，目标字及其上下文中的字均具有各自的原始数值。Attention 机制将目标字作为查询，将其上下文的各个字作为关键字，并通过计算查询与关键字之间的相似性得到权重。这些权重用于将上下文中各个字的数值融入目标字的原始数值中，以实现对语义表示的增强。

为了对输入文本的每个字分别增强语义向量表示，分别将每个字作为 Query，加权融合文本中所有字的语义信息，得到各个字的增强语义向量。在这种情况下，Query、Key 和 Value 的向量表示均来自于同一输入文本，因此，该 Attention 机制也叫 Self-Attention[14]。其输入和输出在形式上完全相同，即文本中各个字的原始向量表示。输出则为经过考虑全文语义信息后各个字融合的增强向量表示。因此，可以将 Multi-head Self-Attention 理解为对文本中每个字分别增强其语义向量表示的一种黑盒机制。本研究中的 Transformer 编码层还需要在多头注意力模块中再添加 3 种关键操作：

(1)残差连接[15]：为实现最后的输出，将模块的输入与输出直接相加。

(2)Layer Normalization[16]：对某一层神经网络节点做均值为 0、方差为 1 的标准化。

(3)线性转换[17]：为提升整体模型的表达能力，对每个字的增强语义向量进行 2 次线性变换。原始向量经过 2 次线性变换，确保变换后的向量与原向量在长度上保持一致。其中，第一次线性变换($\boldsymbol{Q}$，$\boldsymbol{K}$，$\boldsymbol{V}$) 用于计算查询、键和值向量，数学公式具体如下：

$$\boldsymbol{Q} = \boldsymbol{X}\boldsymbol{W}^{Q},\quad \boldsymbol{K} = \boldsymbol{X}\boldsymbol{W}^{K},\quad \boldsymbol{V} = \boldsymbol{X}\boldsymbol{W}^{V} \tag{1}$$

其中，$\boldsymbol{W}^{Q}$，$\boldsymbol{W}^{K}$，$\boldsymbol{W}^{V}$ 表示学习权重矩阵，用来将输入 $\boldsymbol{X}$ 分别映射为查询、键和值向量。第二次线性变换则将多个注意力头的输出拼接，并映射到最终维度。其公式如下：

$$\begin{aligned}&\mathrm{head}_i = \mathrm{Attention}(Q_i, K_i, V_i)\\&\mathrm{Concat}(\mathrm{head}_1,\cdots,\mathrm{head}_h) = \mathrm{Concat}(\mathrm{head}_1,\cdots,\mathrm{head}_h)\boldsymbol{W}^{O}\end{aligned} \tag{2}$$

其中，$\boldsymbol{W}^{O}$ 表示另一个学习矩阵。

2 次变换中，学习的权重矩阵允许模型自适应地学习不同的表示，从而更好地捕捉输入序列的复杂语义结构。同时，不同注意力头之间的并行计算和学习能够增强彼此之间的互补性。每个注意力头可能会关注输入序列中的不同部分或特征，通过将其输出拼接在一起，模型可以综合利用这些不同的表示，从而得到更丰富和全面的语义信息。

Transform Encoder 内部结构如图 2 所示。由图 2 可以看到，Transformer Encoder 的输入和输出在形式上是完全相同的，因此，Transformer Encoder 同样可以表示为将输入文本中各个字的语义向量转换为相同长度的增强语义向量。

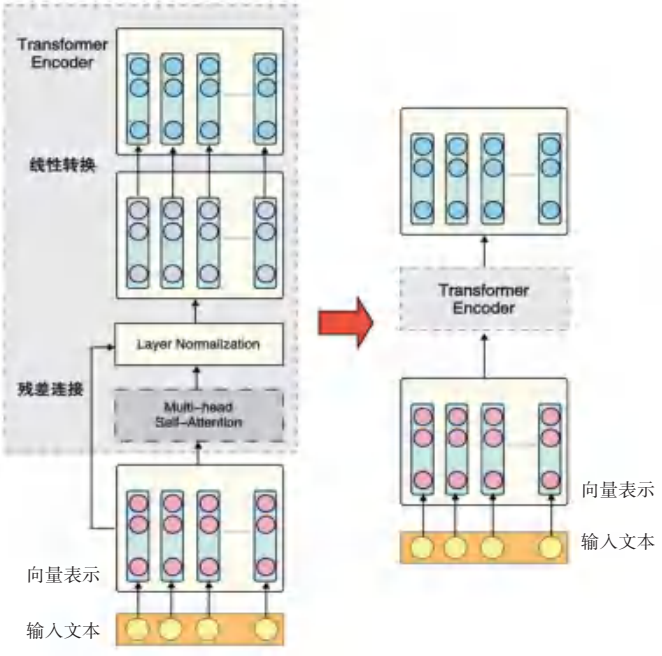

图 2　Transformer Encoder 内部结构图

Fig. 2　Internal structure diagram of Transformer Encoder

BERT 模型是由 12 个 Transformer Encoder 堆叠而成。当然,在更复杂的模型中,也有堆叠 24 层的模型。在本文模型 BPFP 中,经过实验,3 层 Transformer Encoder 堆叠而成已经达到了最好的效果。

这一结果表明,在特定的应用场景和数据集上,过度复杂的模型结构并不总是必要的。通过减少 Transformer Encoder 层数,BPFP 不仅保持了模型的有效性,而且显著提高了计算效率。这种简化方法不仅减少了训练和推断过程中的计算负担,也降低了模型对硬件资源的需求,使其更适合在资源受限的环境中部署。

1.3 BERT 层

在本文模型 BPFP 的架构中,BERT 层首先负责定义模型的关键参数,包括 Transformer 块的数量、词嵌入的维度等。接下来,模型初始化嵌入层 self.Embedding,以及指定数量的 Transformer 编码器。

BERT 层还规定了数据处理的方法:首先,输入数据通过嵌入层,得到嵌入层的总输出。然后,这些输出数据会依次经过一系列 Transformer 模块。在这些模块的作用下,浅层的词嵌入被转换成更深层次的词嵌入。

编码完成后,BERT 编码的每个批次(batch)中第一个字的编码会被送入用于 NSP 任务的全连接层。经过全连接层处理后,输出的结果是预测这一批次输入中每条样本是否为连续的 2 句话的 logits(经过 SoftMax[18] 激活函数转化为概率)。另一方面,编码的输出 x 与嵌入层中 token_embedding 的字对应词嵌入表进行转置后的相乘,用于预测序列中每一个位置对应词表中的哪个字。

1.4 提示学习模块

prompt 主要思想是通过设计提示模板,实现使用更少量的数据在预训练模型上得到更好的效果。提示学习模块从人工构建模板(PET)到 Prefix Tuning[19] 后、又到 P-tuning[20],再到后来的 P-tuning-v2。本文基于 P-tuning 提出一种自动构建的提示模板的方法,利用连续的可学习的向量构建伪提示,模型利用这些连续向量来指导模型训练过程中每个词,通过梯度下降的方式自动构建出有效的提示文本模板。

首先定义语言模型为 M,将输入样本定义为 $H=\{h_0,h_1,h_2,\cdots,h_n\}$,将输入样本对应的标签定义为 $y \in Y$,这里 Y 表示所有标签类别,可训练的伪模版定义为 $P=\{p_0,p_1,p_2,\cdots,p_m\}$。完整的输入模板 X_T 可表示为:

$$X_T=\{K(H),K(P),K(Y)\} \tag{3}$$

其中,$K(H)$ 表示输入样本 H 的词嵌入;$K(Y)$ 表示标签词 Y 的词嵌入;$K(P)$ 表示可训练的伪模板的词嵌入。

在定义伪模版时,由于 BERT 词表中预留的特殊符号[unused]本身并无实际含义,任意组合对语义影响不大。因此,直接使用[unused]作为构建伪提示模板的元素。但是组成伪提示模板的这些[unused]之间是离散的。为了将这些离散的[unused]链接起来,本文使用多层感知机作为提示编码器,将 BERT 词表中的[unused]建模为一个序列,并参与训练。此外,在训练过程中,利用下游损失函数 Λ 并采用随机梯度下降的方法,对伪模板的参数进行连续优化,以找到最好的伪模板,求解公式为:

$$K(T)=\arg_T\min\Lambda(M(K(H),K(Y))) \tag{4}$$

这种自动构建模板的方法可以实现更加智能的模型。图 3 为本文使用的提示学习模板。

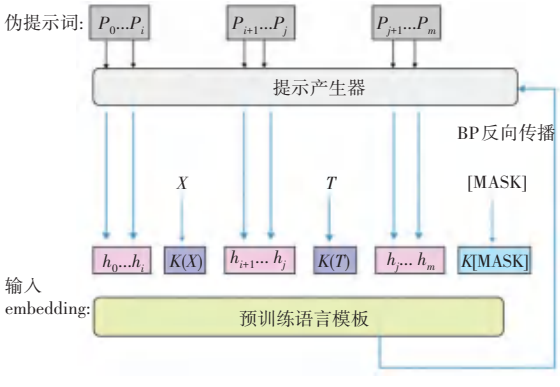

图 3 提示学习模板

Fig. 3 Prompt learning template

2 实验与分析

2.1 实验数据和评价指标

实验数据来自《英雄联盟》宇宙,内容主要是英雄联盟背景故事,共约 22 万字。

使用预测下一句话的准确率、遮挡语言的准确率、损失函数收敛曲线、预测时间等客观指标来评价模型的性能。数学定义如下:

$$\text{Mlm\_acc}=\frac{\text{Num}_{\text{acc}}}{\text{Num}_{\text{tot}}}\times 100\% \tag{5}$$

其中,Mlm_acc 表示遮挡语言模型准确率;Num_{acc} 表示正确预测的单词数;Num_{tot} 表示总遮蔽数。同时,研究中又推得公式如下:

$$\text{Nsp\_acc}=\frac{\text{Num}_{\text{cou}}}{\text{Num}_{\text{adj}}}\times 100\% \tag{6}$$

其中,Nsp_acc 表示预测下句话准确率;Num_{cou}

表示正确预测的相邻对数；Num_{adj} 表示总的相邻对数。

BPFP 模型的损失由 2 部分组成：MLM 损失和 NSP 损失。总损失函数可以表示为两者之和。数学定义如下：

$$L_{MLM} = -\frac{1}{N}\sum_{i=1}^{N}\log P(W_a \mid W_c) \tag{7}$$

$$L_{NSP} = -\log P(y \mid adj) - \log P(1-y \mid non\_adj) \tag{8}$$

其中，W_a 表示实际词；W_c 表示上下文；adj 表示相邻；non-adj 表示不相邻。

最终 BPFP 模型的总损失函数为两者之和，即：

$$L_{总} = L_{MLM} + L_{NSP} \tag{9}$$

2.2　结果分析

2.2.1　实验结果

实验结果如图 4 所示，其中 Mlm_acc、Nsp_acc、Mlm_loss、Nsp_loss 分别表示遮挡语言模型准确率、预测下一句的准确率、遮挡语言模型的损失函数以及预测下一句的损失函数，loss 表示整个 BPFP 预训练模型的总损失。

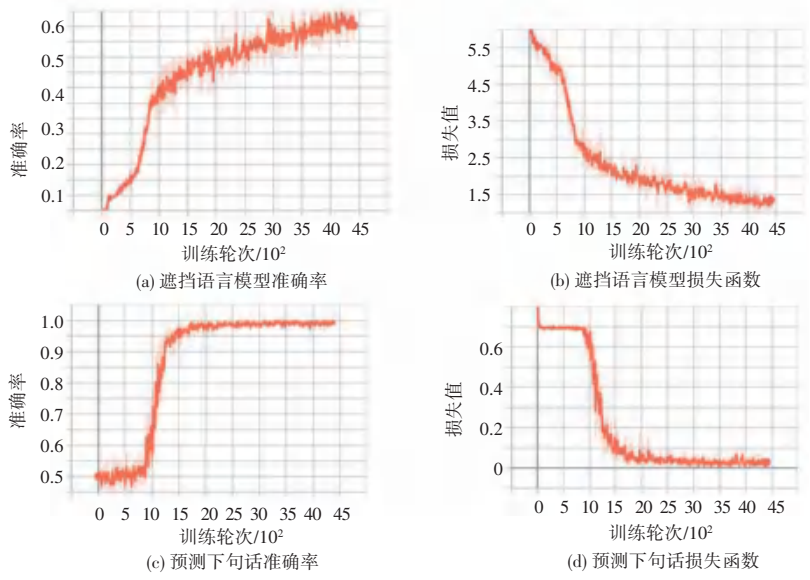

图 4　实验结果图

Fig. 4　Diagram of experimental results

实验结果表明 BPFP 模型在经过一段时间训练后，遮挡语言模型的准确率收敛在 70%左右，下一句话预测准确率接近 1。同时，两者的损失函数经过一段时间的训练后，分别收敛于 1.2 和 0。

2.2.2　对比试验

为了进一步验证 BPFP 模型的训练效果，展开了对比试验。分为 2 个部分，分别是针对长文本和短文本的预测结果分析，在 2 种不同的数据集下，与当下几种主流预训练模型从精度、损失函数、训练时间等方面做比较，分别是改进的双向 RNN、Bert、Word2vec + LDA、BERT + CNN 、ChpoBERT 以及 BERT+BiLSTM。首先，是基于长文本的模型评测，实验结果见表 1。

根据表 1 的实验结果，本文的 BPFP 模型相较于初始 Bert 模型，在预测遮挡模型和下一句话的准确率上分别提升了 6.9% 和 3.7%，相较于改进的 Bert 模型来说，在准确率上的提升并不明显。然而，本文模型 BPFP 由于内部仅有 3 层 Transformer 层，训练同一个数据集的时间只有其它模型的 1/3。

短文本数据集的仿真实验结果见表 2。从短文本数据集的实验结果中可知，基于基础模型进行微调的方法并未改变原有模型的结构。然而，在处理小文本数据时，这些较大的模型往往未能展现出预期的训练效果。与此相反，本研究提出的 BPFP 模型在处理遮挡语言预测任务时，准确率提升显著、提升值为 13.5%。同时，与其他基于 BERT 微调模型如 BERT+ CNN、BERT + biLSTM、ChpoBERT 相比，BPFP 模型的准确率分别提升了 12.3%、7.1%、5.9%。在预测下一句话的准确率方面，BPFP 模型无论在长文本、还是短文本数据集上，均有出色表现，准确率分别达到了 99.2%和 99.5%。

表 1 长文本数据集模型表现

Table 1 Performance of models on long text datasets

模型	Mlm_acc/%	Nsp_acc/%	Mlm_loss	Nsp_loss	loss
双向 RNN	48.2	88.9	-	-	-
Word2Vec+LDA	50.1	86.8	-	-	-
BERT	65.0	95.5	3.20	1.45	4.65
BERT+CNN	66.2	96.0	1.98	0.96	2.94
BERT+BiLSTM	68.4	96.0	2.05	1.85	3.90
ChpoBERT	72.6	98.8	1.90	1.23	3.13
BPFP	71.9	99.2	1.09	0.01	1.10

表 2 短文本数据集模型表现

Table 2 Performance of models on short text datasets

模型	Mlm_acc/%	Nsp_acc/%	Mlm_loss	Nsp_loss	loss
双向 RNN	48.4	87.2	-	-	-
Word2Vec+LDA	55.1	88.2	-	-	-
BERT	69.0	92.3	2.20	1.15	3.35
BERT+CNN	70.2	93.0	1.08	0.90	1.98
BERT+BiLSTM	75.4	96.0	1.36	1.05	2.41
ChpoBERT	76.6	96.8	1.36	1.03	2.39
BPFP	**82.5**	99.5	0.61	0.03	0.64

此外,BPFP 模型的损失函数,相比于原始 BERT 模型及已经进行微调的 BERT 模型,得到了较小值。特别是与表现最好的 BERT+CNN 模型相比,BPFP 的损失函数值减少了一半。这一结果表明 BPFP 模型具有更强的鲁棒性,能够在各种文本处理任务中更有效地优化并减少错误。

3 结束语

在自然语言处理领域,预训练模型得到了广泛应用。针对传统预训练模型在训练效果和改进模型复杂度方面存在的问题,本文提出了一种创新方法,将提示学习与微调模型相结合。本文模型在长文本预测方面相较于传统模型有了明显的提升,虽然相对于改进的 Bert 模型,精度提升并不明显,但在大幅度缩短模型预训练时间方面取得了显著成效。在短文本训练方面,本文提出的 BPFP 模型则展现出显著的优势。未来的研究中,针对不同的 NLP 任务,可以设计更具体的提示学习模板以进一步提升模型效果。这一研究为提高自然语言处理模型的性能和效率提供了有益的参考。

参考文献

[1] LINA S. Predicting the risk of Myopia exacerbation based on a Naive Bayesian classification algorithm[C]//Proceedings of 2023 IEEE Ural - Siberian Conference on Biomedical Engineering, Radioelectronics and Information Technology (USBEREIT). Piscataway, NJ: IEEE, 2023: 68-71.

[2] CHAVALI S T, KANDAVALLI C T, SUGASH T M, et al. Grammar detection for sentiment analysis through improved Viterbi algorithm[C]//Proceedings of 2022 International Conference on Advances in Computing, Communication and Applied Informatics (ACCAI). Piscataway, NJ: IEEE, 2022: 1-6.

[2] PUERTO - SANTANA C E, LARRAñAGA P, BIELZA C. Feature saliencies in asymmetric hidden Markov models[J]. IEEE Transactions on Neural Networks and Learning Systems, 2024, 35(3): 3586-3600.

[4] WANG Jianping, ZHANG Xueyan, GAO Guohong, et al. Open pose mask R-CNN network for individual cattle recognition[J]. IEEE Access, 2023, 11: 113752-113768.

[5] MA Z, ZHANG H, LIU J. DB-RNN: An RNN for precipitation nowcasting deblurring[J]. IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing, 2024, 17: 5026-5041.

[6] GAO Shuai, ZHANG Shuo, HUANG Yuefei, et al. A new Seq2Seq architecture for hourly runoff prediction using historical rainfall and runoff as input[J]. Journal of Hydrology, 2022, 612: 128099.

[7] MUTINDA J, MWANGI W, OKEYO G. Sentiment analysis of text reviews using lexicon-enhanced bert embedding (LeBERT) model with convolutional neural network[J]. Applied Sciences, 2023, 13(3): 1445.

[8] CHEN X, CONG P, LV S. A long-text classification method of Chinese news based on BERT and CNN[J]. IEEE Access, 2022, 10: 34046-34057.

[9] YANG Linda, HUANG Baohua, GUO Shiqian. A small-sample

text classification model based on pseudo-label fusion clustering algorithm[J]. Applied Sciences, 2023, 13: 4716.
[10]MA Qinghua, DONG Ming, XIA Changjie, et al. A multivariate normal distribution data generative model in small-sample-based fault diagnosis: Taking traction circuit breaker as an example [J]. IEEE Transactions on Intelligent Transportation Systems, 2024, 25(6):5825-5841.
[11]LIU X, JI K, FU Y, et al. P-tuning v2: Prompt tuning can be comparable to fine-tuning across scales and tasks[J]. arXiv preprint arXiv,2110.07602,2021.
[12]贾钰峰,李容,章蓬伟,等. 基于字向量的短文本情感分类研究[J]. 微处理机,2023,44(6):40-45.
[13]CHEN H, WU G D, LI J X, et al. Research advances on deep learning recommendation based on attention mechanism [J]. Computer Engineering & Science, 2021, 43(2): 370-380.
[14]LEE S, HWANG R, PARK J, et al. HAMMER: Hardware-friendly approximate computing for self-attention with mean-redistribution and linearization[J]. IEEE Computer Architecture Letters, 2023, 22(1): 13-16.
[15]ZHAO Jingjiao, ZHAO Zhihong, YANG Shaopu. Rolling bearing fault diagnosis based on residual connection and 1D-CNN [J]. Journal of Vibration and Shock, 2021, 40(10): 1-6.
[16]HUANG Lei, QIN Jie, ZHOU Yi, et al. Normalization techniques in training dnns: Methodology, analysis and application[J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2023,45(8): 10173-10196.
[17]ZHU Xinhua, ZHU Yuxiang, ZHANG Lanfang, et al. A BERT-based multi-semantic learning model with aspect-aware enhancement for aspect polarity classification [J]. Applied Intelligence, 2023, 53(4): 4609-4623.
[18]姜健,魏小源. 自适应注意力 LSTM-ResNet 下的滚动轴承故障诊断[J]. 制造技术与机床,2024(6):74-81.
[19]郭新浩. 基于 Prompt 的文本生成技术研究与实现[D]. 北京:北京邮电大学,2023.
[20]于碧辉,蔡兴业,魏靖烜. 基于提示学习的小样本文本分类方法[J]. 计算机应用,2023,43(9):2735-2740.