计算机科学
COMPUTER SCIENCE

融合抗噪和双重蒸馏的文本分类方法

郭伟，黄嘉晖，侯晨煜，曹斌

引用本文

郭伟，黄嘉晖，侯晨煜，曹斌.融合抗噪和双重蒸馏的文本分类方法[J]．计算机科学, 2023, 50(6): 251-260.

GUO Wei, HUANG Jiahui, HOU Chenyu, CAO Bin. Text Classification Method Based on Anti-noise and Double Distillation Technology [J]. Computer Science, 2023, 50(6): 251-260.

计算机科学 COMPUTER SCIENCE
http://www.jsjkx.com
DOI:10.11896/jsjkx.220500100

融合抗噪和双重蒸馏的文本分类方法

郭 伟 黄嘉晖 侯晨煜 曹 斌
浙江工业大学计算机科学与技术学院 杭州 310023
(weiguo1014@zjut.edu.cn)

摘 要 文本分类是自然语言处理中重要且经典的问题,常被应用于新闻分类、情感分析等场景。目前,基于深度学习的分类方法已经取得了较大的成功,但在实际应用中仍然存在以下3个方面的问题:1)现实生活中的文本数据存在大量的噪声标签,直接用这些数据训练模型会严重影响模型的性能;2)随着预训练模型的提出,模型分类准确率有所提升,但模型的规模和推理计算量也随之提升明显,使得在资源有限的设备上使用预训练模型成为一项挑战;3)预训练模型存在大量的冗余计算,当数据量较大时会导致模型出现预测效率低下的问题。针对上述问题,提出了一个融合抗噪和双重蒸馏(包括知识蒸馏和自蒸馏)的文本分类方法,通过基于置信学习的阈值抗噪方法和一种新的主动学习样例选择算法,以少量的标注成本提升数据的质量。同时,通过知识蒸馏结合自蒸馏的方式,减小了模型规模和冗余计算,进而使其可以根据需求灵活调整推理速度。在真实数据集上进行了大量实验来评估该方法的性能,实验结果表明所提方法在抗噪后准确率提升了1.18%,在较小的精度损失下相比BERT可以加速4~8倍。

关键词:噪声标签;置信学习;主动学习;知识蒸馏;自蒸馏
中图法分类号 TP391

Text Classification Method Based on Anti-noise and Double Distillation Technology

GUO Wei, HUANG Jiahui, HOU Chenyu and CAO Bin
College of Computer Science & Technology, Zhejiang University of Technology, Hangzhou 310023, China

Abstract Text classification is an important and classic problem in the field of natural language processing, and it is often used in news classification, sentiment analysis and other scenarios. The existing deep learning-based classification methods have the following three problems: 1) There are a large number of noisy labels in real-life datasets, and directly using these data to train the model will seriously affect the performance of the model. 2) With the introduction of the pre-training model, the accuracy of model classification has improved, but the scale of the model and the number of inference calculations have also increased significantly, which make it a challenge to use pre-training models on devices with limited resources. 3) The pre-training model has a large number of redundant calculations, which will lead to low prediction efficiency when the amount of data is large. To address these issues, this paper proposes a text classification method that combines anti-noise and double distillation (including knowledge distillation and self-distillation). Through the threshold anti-noise method based on confidence learning and a new active learning sample selection algorithm, the quality of the data is improved with a small amount of labeling cost. Meanwhile, the combination of knowledge distillation and self-fistillation reduces the scale of the model and redundant calculation, thereby it can flexibly adjust the inference speed according to the demand. Extensive experiments are performed on real datasets to evaluate the performance of the proposed method. Experimental results show that the accuracy of the proposed method after anti-noise increases by 1.18%, and it can be 4~8 times faster than BERT under small accuracy losses.

Keywords Noise label, Confidence learning, Active learning, Knowledge distillation, Self-distillation

到稿日期:2022-05-12 返修日期:2022-10-12
基金项目:国家自然科学基金(62276233);浙江省重点研发计划(2022C01145)
This work was supported by the National Natural Science Foundation of China (62276233) and Key R&D Program of Zhejiang Province (2022C01145).
通信作者:侯晨煜(houcy@zjut.edu.cn)

1 引言

文本分类是自然语言处理中的一个经典问题[1]，即根据原文本的内容或主题，按照既定的分类标准得到文本相应的类别。如今处于大数据时代不断发展的阶段，互联网上各行业领域内的文本数据呈现爆炸式增长的趋势，文本包含的语义信息也更加丰富，而文本分类可以帮助各行业有效地利用各个领域中的文本语义信息，挖掘文本中蕴含的潜在价值信息。例如，在电信行业的投诉服务中，客户会与通过客服通话来表达投诉的问题，通常会将通话内容转为文本形式，其中会涵盖多个业务方面的诉求，如宽带故障、话费问题等。文本的复杂性使电信客服难以准确区分投诉内容所属类别，而文本分类方法能有效解决此问题，以便进一步处理。

近年来，深度学习嵌入模型被广泛地应用到了文本分类任务中，并取得了较好的效果。例如，谷歌开发的一系列word2vec模型[2]、Transformer网络架构[3]、以及由340×10^6参数组成的BERT模型[4]，都极大地提升了模型分类的准确率。虽然目前的文本分类方法训练的准确率在不断提升，但由于现有的文本分类方法越来越趋向于使用更大的模型和更多的训练数据，因此它们的不足之处也逐渐显现。在实际应用中，文本分类方法面临着三大挑战。

(1)噪声数据问题。噪声即数据集中被人工标注错误的样本，噪声的存在是真实数据中的一个严峻且常见的问题。在文本分类领域中，对大量噪声文本的预处理会浪费大量人力与物力，还会造成新的标注错误。而现有的技术越来越依赖于大量数据的训练，而噪声数据的存在使得训练所得模型准确率低、鲁棒性差。

(2)受限资源环境下的模型部署。在文本分类模型构建时会加入预训练语言模型来提高模型的分类准确率。随着模型参数量不断增加，计算量、占用的内存空间也会相应增长。而当模型需要部署到资源有限的终端时，上述问题将会极大地影响模型的部署。

(3)效率问题。预训练模型具有冗余的计算[5]，当数据量大幅度增加时，运行时间也会不断增加，效率逐渐降低，无法满足含大规模数据且实时性要求高的应用。

针对上述问题，本文提出了一种融合抗噪和双重蒸馏的文本分类方法。现有的处理噪声数据的技术分为两类：筛选出噪声数据后再重新训练模型和直接基于噪声数据训练。第一类方法中，典型的置信学习[6]方法通过先过滤错误样本再进行重新训练来提升模型性能。但是现实场景中的数据往往是不平衡的，即每个类别的样本数差距很大。而样本数少的类别直接进行噪声过滤会进一步加剧不平衡问题，使得模型无法对样本数少的类别进行很好的训练。对此，我们更希望能改正样本数较少类别的标注错误。第二类方法中，基于半监督学习[7]的方法利用少量带标签样本和大量无标签样本来训练模型并给无标签数据打上伪标签。而真实生活中产生的数据往往包含了多个类别，即使没有任何噪声数据，模型的分类准确率也不高。因此，该方法用低质量的伪标签训练模型对分类效果的提升很小。针对上述两种方法的不足，本文在噪声数据问题上，提出了一种结合置信学习和主动学习[8]的动态抗噪方法，通过置信学习筛选出文本中的噪声数据，利用样本数量阈值动态控制噪声删除，并利用Self-Confidence[6]和reBvSB[8]的主动学习样例选择算法，筛选少量有价值的噪声数据用于主动学习中的专家标注，降低标注成本。本文方法在保证数据平衡的条件下，使用较少的标注成本，提高了数据集的标注质量。

在模型部署问题上，现有的许多研究通过缩减模型大小来方便模型部署，提高模型的实用性，如权重剪枝[9]、参数共享[10]和知识蒸馏[11]。其中，知识蒸馏是目前流行且实用的方法[11]。它将完成训练的复杂模型(教师模型)中的知识经过蒸馏迁移到简单模型(学生模型)中，将教师模型输出的类别概率作为软目标来训练学生模型，使学生模型具有教师模型的泛化能力。模型经过压缩后变小，参数量也会减少，达到了可接受的速度和精度平衡的要求，但由于该方法中模型固定，因此仍存在计算冗余问题，无法应对数据量急剧增加的情况。而Liu等提出的FastBERT[12]方法，因结合了独特的样本自适应机制和自蒸馏，能很好地解决数据量快速变化的问题。该方法不断训练自身模型中的多层学生分类器，并通过置信度和阈值进行自适应推理，以判断样本是否预测正确，若预测正确，样本即可在当前分类器输出，否则需要通过更多层分类器预测。但是训练后达到较好效果的FastBERT具有12层Transformer结构，模型较大，无法解决部署困难问题。为了可以在有限环境下部署模型并使模型具有自适应能力，本文提出了一种知识蒸馏[10]和自蒸馏[12]相结合的双重蒸馏方法，首先通过知识蒸馏将12层Transformer的复杂模型压缩成3层Transformer的小模型，再基于小模型运用自蒸馏和自适应机制，动态地调整样本执行层数，减少模型的冗余计算，实现高效且方便部署的文本分类模型。

本文的主要贡献如下：

(1)提出了结合置信学习与主动学习的动态抗噪方法。通过置信学习筛选出噪声数据，比较样本数量阈值和类别样本数动态地删除噪声，使删除噪声后的样本数满足样本数量阈值以保证数据平衡。在此基础上通过Self-Confidence[6]和reBvSB[8]主动学习样例选择算法，准确地过滤掉孤立点，筛选出数据集中最有价值的噪声样本并以少量的人工标注成本，高效地提升数据集的样本质量。

(2)提出了知识蒸馏和自蒸馏相结合的双重蒸馏方法。我们利用知识蒸馏将12层Transformer的复杂模型压缩成3层Transformer的小模型，并在小模型的基础上使用自蒸馏和自适应推理，不断地训练每层Transformer后的分类器，当样本的预测置信度满足预测阈值时即可输出，否则继续训练。本方法不仅压缩了模型，提升了模型的训练和预测速度，还使模型具有可以灵活调整的推理速度，进而使得模型在业务中更好地投入使用。

(3)在真实数据集上进行了大量实验，结果证明本模型具有良好的抗噪能力和高效的预测性能。在经过文本抗噪后，

模型的准确率提升了1.18%，同时本模型在较少的精度损失下，预测阶段可以加速4～8倍。

本文第2节介绍了文本抗噪、模型压缩和加速的常用方法；第3节概述了模型的各个模块；第4节通过实验分析了各个模块的性能；最后总结全文。

2　相关工作

本文的相关工作可以分为两类：1）文本抗噪；2）模型压缩和加速技术。

2.1　文本抗噪

目前处理噪声的方法主要分为两类。第一类方法是识别并去除数据集中的噪声数据后进行重新训练，这样可以大大减少噪声数据对模型性能的影响。Northcutt等[6]提出置信学习框架来提取噪声数据，进而在除噪后的干净数据基础上重新训练模型，达到提高模型准确率的效果。第二类方法是利用噪声数据进行学习。Tanaka等[13]提出了一个学习模型参数并估计真实标签的联合优化框架。Zhang等[7]采用数据增强的方式扩大干净数据集，并对噪声数据集进行标签猜想，取得了很好的效果。不同于以上的研究，本文采用置信学习结合主动学习的方式来处理噪声样本。首先我们不希望放弃噪声样本，其次直接用噪声数据来训练往往需要一份干净的数据，这在现实世界中很难实现。

2.2　模型压缩与加速技术

预训练语言模型，如BERT[4]，已经在很多领域取得了很好的成绩，由于模型规模太大，很难将其部署在资源有限的环境中，因此可以采用模型压缩技术来提高模型的实用性。目前主流的压缩技术主要有五大类：参数剪枝[5]、参数共享[9]、低秩分解[14]、模型量化[15]和知识蒸馏[10]。本文的工作属于知识蒸馏方向，目的是减小模型的规模，加快推理速度。知识蒸馏可以分为两种方式。1）训练一个完备的大模型，将大模型的知识迁移到额外的小模型中训练。这是目前研究较多的知识蒸馏方式。TinyBERT[16]提出了一种两阶段的蒸馏框架，分别在预训练和微调过程中进行Transformer蒸馏。Sun等[17]提出了两种耐心学习策略，使学生模型可以学习教师网络的中间层。但这些方法的模型是固定的，无法解决计算冗余等问题。2）自蒸馏，即知识迁移的对象是自己。Liu等提出了一个FastBERT[12]模型，首次将样本自适应机制和自蒸馏相结合。但是，这种方法在训练阶段效果并没有提升，且当效果较好时，模型也较大。与上述研究不同，本文提出了一种知识蒸馏和自蒸馏结合的双重蒸馏方法，使用小模型解决了训练速度较慢的问题并使用自适应推理机制解决了计算冗余问题。

3　方法概述

本节将介绍融合抗噪和双重蒸馏的文本分类方法。本方法不仅可以动态删除文本中的噪声标签并选出最有价值的噪声样本进行人工标注，还可以进行模型压缩和加速，使样本具有自适应推理能力。本模型主要分为3个模块：数据抗噪模块、主干网络训练模块以及模型压缩和加速模块。模型的框架如图1所示，下面将对各个模块进行介绍。

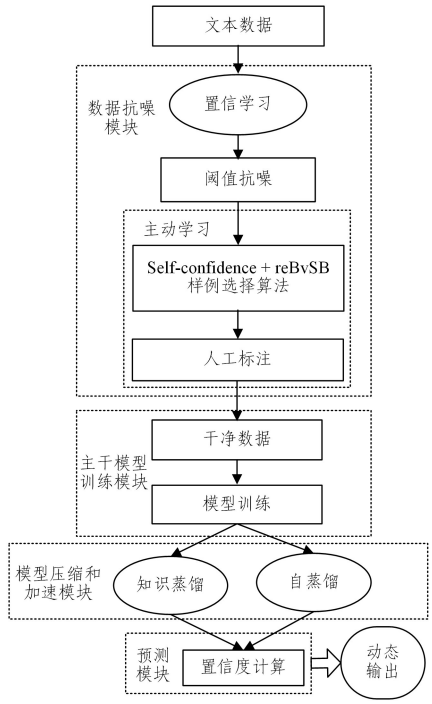

图1　模型框架图

Fig. 1　Model frame diagram

3.1　数据抗噪模块

在以数据驱动的深度学习模型训练中，训练数据的质量直接影响了模型的泛化能力。然而，在现实世界中产生的真实数据往往包含大量噪声，为了避免噪声数据对模型产生负面作用，本文提出了一种结合置信学习和主动学习样例选择算法的动态抗噪方法。首先，运用置信学习估计数据集 D 中人工标注的标签 $\tilde{y}$（杂乱的）和潜在的正确标签 y^*（未知的）之间的联合分布，进而筛选出噪声样本。在筛选出噪声样本后，由于多种类别之间的样本数量不平衡，直接删除各类别中的噪声样本会加重不平衡问题，甚至会使原本低样本量的类别从数据集中消失，影响模型的训练。因此，我们设置了样本数量阈值 λ 来动态筛选噪声标签样本，若删除噪声样本后，该类别的样本数满足样本数量阈值条件即可删除噪声样本；若删除样本后不能满足该阈值，可根据主动学习样例选择算法，选择一定量的有价值样本进行人工标注，在保证样本总数满足该阈值的前提下，进一步提高数据质量。具体分为以下3个步骤。

步骤1　采用置信学习[6]方法，筛选出噪声标签样本。

（1）将带有噪声标签的样本 x 输入参数为 $\boldsymbol{\theta}$ 的模型中，并计算第 h 个样本属于第 j 个类别的概率 $p(\hat{y}=j;\boldsymbol{x}_h,\boldsymbol{\theta})$。其中，$j=1,\cdots,m$，$m$ 表示类别总数；$h=1,\cdots,n$，n 表示样本总数。在后面的公式中，我们将 $p(\hat{y}=j;\boldsymbol{x}_h,\boldsymbol{\theta})$ 简写为 $\hat{p}_j$。本步骤伪代码如算法1第1－7行所示。

示例：如图2所示，假设数据集中包含3个类别，分别记作“1”“2”“3”，各类样本数分别为5，8，11，总样本数为24。图中 $\hat{p}_j$ 表示 $\boldsymbol{x}_h$ 样本经过模型预测第 j 个类别的概率。

$\hat{y}=j$ / $\hat{p}_j$ / x_h	1	2	3	$\tilde{y}$	y^*
x_1	0.6	0.3	0.1	1	无
x_2	0.7	0.1	0.2	1	1
x_3	0.8	0.1	0.1	1	1
x_4	0.4	0.6	0.0	1	2
x_5	0.9	0.1	0.0	1	1
x_6	0.2	0.7	0.1	2	2
x_7	0.2	0.5	0.3	2	2
x_8	0.2	0.6	0.2	2	2
x_9	0.1	0.1	0.8	2	3
x_{10}	0.1	0.8	0.1	2	2
x_{11}	0.0	0.9	0.1	2	2
x_{12}	0.4	0.2	0.4	2	无
x_{13}	0.7	0.2	0.1	2	1
x_{14}	0.2	0.2	0.6	3	无
x_{15}	0.1	0.2	0.7	3	3
x_{16}	0.1	0.1	0.8	3	3
x_{17}	0.1	0.0	0.9	3	3
x_{18}	0.0	0.6	0.4	3	2
x_{19}	0.1	0.0	0.9	3	3
x_{20}	0.1	0.0	0.9	3	3
x_{21}	0.0	0.1	0.9	3	3
x_{22}	0.7	0.1	0.2	3	1
x_{23}	0.1	0.6	0.3	3	2
x_{24}	0.1	0.1	0.8	3	3
t_j	0.68	0.50	0.67		

图 2 示例图

Fig. 2 Sample graph

(2)我们需要为每个类别设置一个噪声阈值,进而筛选出每个类别中满足噪声阈值的样本。其中,噪声阈值 t_j 的计算式如式(1)所示:

$$t_j=\frac{1}{|\boldsymbol{X}_{\tilde{y}=j}|}\sum_{x\in\boldsymbol{X}_{\tilde{y}=j}}p(\hat{y}=j;\boldsymbol{x},\boldsymbol{\theta}) \tag{1}$$

其中,$p(\hat{y}=j;x,\boldsymbol{\theta})$,$x\in\boldsymbol{X}_{\tilde{y}=j}$ 表示样本人工标注的类别为 j,且预测类别也为 j 的概率,即自置信度;$|\boldsymbol{X}_{\tilde{y}=j}|$ 表示人工标注为 j 类别样本的数量。噪声阈值 t_j 是由每个类别样本的所有自置信度求平均值所得。其伪代码如算法 1 第 8－15 行所示。

示例:如图 2 所示,噪声阈值 t_j 是对每个类别中所有预测为原标签类别的概率求平均值得到,即对图中 3 个灰色块分别计算其平均值得到 3 个类别的噪声阈值,分别为 0.68,0.5,0.67。

(3)将 $\hat{p}_j$ 与相应类别的噪声阈值 t_j 进行比较,判断该样本的正确标签 y^*,计算式如式(2)所示:

$$y^*=\{\arg\max\hat{p}_j\mid\hat{p}_j\geqslant t_j;j\in[m]\} \tag{2}$$

我们对每一个样本进行如下计算:将其预测为每个类别的概率 $\hat{p}_j$ 与相应类别的噪声阈值 t_j 进行比较,从大于阈值的概率中选取最大概率所对应的类别 j 作为该样本的潜在正确标签。其伪代码如算法 1 第 16－18 行所示。

示例:样本的潜在正确标签结果如图 2 中 y^* 一列所示。计算样本正确类别包括两种情况,详细示例如下。

情况 1 样本的 3 个类别预测概率存在大于相应类别的噪声阈值的概率 $\hat{p}_j$。例如,x_2 样本的 3 个类别的预测概率为 0.7,0.1,0.2,分别与噪声阈值 0.68,0.5,0.67 进行比较,其中只有 $\hat{p}_1$ 大于 t_1,因此,x_2 样本的正确类别为 1。

情况 2 样本的 3 个类别预测概率中不存在大于相应类别的噪声阈值的概率 $\hat{p}_j$。例如,x_1 样本的 3 个类别的为 0.6,0.3,0.1,分别与噪声阈值 0.68,0.5,0.67 进行比较,预测概率均小于对应的噪声阈值,因此,x_1 样本的正确类别为空,直接判定为噪声数据,不加入后续的联合分布计算。

(4)通过计数统计得到原标签 $\tilde{y}$ 为 i,潜在正确标签 y^* 为 k 的样本个数,进而组成置信联合分布 $\boldsymbol{C}_{\tilde{y}=i,y^*=k}$,计数如式(3)所示:

$$\boldsymbol{C}_{\tilde{y}=i,y^*=k}=|\boldsymbol{X}_{\tilde{y}=i,y^*=k}| \tag{3}$$

其中,$|\boldsymbol{X}_{\tilde{y}=i,y^*=k}|$ 表示原标签 $\tilde{y}$ 为 i,潜在正确标签 y^* 为 k 的样本个数。其伪代码如算法 1 第 19－27 行所示。

示例:如图 3 所示,我们通过相应样本数量统计得到置信联合分布 $\boldsymbol{C}_{\tilde{y}=i,y^*=k}$,图中我们使用 $\boldsymbol{C}[i][k]$ 表示。例如,如图 2 所示,$\boldsymbol{C}[1][1]=3$,即原标签 $\tilde{y}$ 为 1、潜在正确标签 y^* 为 1 的样本数是 3,包括样本 x_2,x_3,x_5。

i / $\boldsymbol{C}[i][k]$ / k	1	2	3
1	3	1	0
2	1	5	1
3	1	2	7

图 3 置信联合分布示例图

Fig. 3 Confidence joint distribution example graph

(5)对 $\boldsymbol{C}_{\tilde{y},y^*}$ 联合分布进行归一化得到估计联合分布 $\hat{\boldsymbol{Q}}_{\tilde{y},y^*}$,其计算式如式(4)所示:

$$\hat{\boldsymbol{Q}}_{\tilde{y},y^*}=\frac{1}{n}\cdot\frac{\boldsymbol{C}_{\tilde{y}=i,y^*=k}}{\sum_{k\in[m]}\boldsymbol{C}_{\tilde{y},y^*=k}}\cdot|\boldsymbol{X}_{\tilde{y}=i}| \tag{4}$$

其中,$|\boldsymbol{X}_{\tilde{y}=i}|$ 表示人工标注为 i 的样本数量,n 表示数据集中的样本总数。首先利用每个原标签 $\tilde{y}$ 为 i、潜在正确标签 y^* 为 k 的样本数 $\boldsymbol{C}_{\tilde{y}=i,y^*=k}$ 除以置信联合分布矩阵 $\boldsymbol{C}_{\tilde{y},y^*}$ 中原标签 $\tilde{y}$ 为 i 的样本总数,再与相应原标签为 i 的样本数量相乘;然后,将所求得的值除以数据集中的样本总数 n,所得矩阵即为归一化后的估计联合分布 $\hat{\boldsymbol{Q}}_{\tilde{y},y^*}$。其伪代码如算法 1 第 28－35 行所示。

示例:如图 4 所示,计算过程中,我们使用 $\boldsymbol{Q}[i][k]$ 表示 $\hat{\boldsymbol{Q}}_{\tilde{y},y^*}$。$\boldsymbol{Q}[1][1]=\frac{1}{24}\cdot\frac{3}{3+1+0}\cdot5\approx0.16$,其余值都采用同样的计算方式。

i / $\boldsymbol{Q}[i][k]$ / k	1	2	3
1	0.16	0.05	0.00
2	0.05	0.24	0.05
3	0.05	0.09	0.32

图 4 估计联合分布示例图

Fig. 4 Estimated joint distribution example graph

(6)由估计联合分布的概率，根据自置信度(Self-Confidence)$\hat{p}(\tilde{y}=i;x\in \boldsymbol{X}_i)$升序排列，在人工标注为$i$的类别样本中选取前$d_{\tilde{y}=i}$个样本，作为该类别的噪声样本$\boldsymbol{X}'_{\tilde{y}=i}$，其计算式如式(5)所示。

$$d_{\tilde{y}=i}=\lceil n\cdot \sum_{k\in 1..m;k\neq i}(\hat{\boldsymbol{Q}}_{\tilde{y}=i,y^*=k})\rceil \tag{5}$$

其中，自置信度表示标签为i被正确打标的概率，n是总样本数。潜在正确标签类别k取不等于i的其他类别。我们需要对估计联合分布$\hat{\boldsymbol{Q}}_{\tilde{y},y^*}$中原标签类别为$i$不等于正确标类别$k$的估计联合分布值求和，然后与样本总数$n$相乘，得到从人工标注为$i$的样本中选取的噪声数量$d_{\tilde{y}=i}$。如果结果不是整数，需要向上取整。其伪代码如算法1第36－46行所示。

算法1　基于置信学习的噪声样本筛选算法

输入：数据集 D＝{x_1，x_2，…x_h，…，x_n}，样本总数 n，类别总数 m

输出：每个类别中的噪声样本$X'_{\tilde{y}=i}$

1. //预测样本的类别概率
2. 初始化模型 Model 的参数 **θ**
3. 用数据集 D 训练模型
4. for $x_h\in$D do
5.　for j←1 to m do
6.　　$p(\hat{y}=j;\mathbf{x}_h,\boldsymbol{\theta})\leftarrow$Model($\mathbf{x}_h$，**θ**)
7.　$\hat{p}_h^j\leftarrow p(\hat{y}=j;\mathbf{x}_h,\boldsymbol{\theta})$
8. //计算每个类别的噪声阈值
9. for j←1 to m do
10.　$\mathbf{X}_{\tilde{y}=j}\leftarrow\varnothing$
11.　for $\mathbf{x}_h\in$D do
12.　　if $\tilde{y}_{x_h}==$j then
13.　　$\mathbf{X}_{\tilde{y}=j}$.add(x_h)
14.　$p_j'\leftarrow\sum_{x_h\in X_{\tilde{y}=j}}p(\hat{y}=j;\mathbf{x}_h,\boldsymbol{\theta})$
15.　$t_j\leftarrow\frac{p_j'}{|X_{\tilde{y}'=j}|}$
16. //计算样本潜在正确标签
17. for $\mathbf{x}_h\in$D do
18.　$y^*_{x_h}\leftarrow\arg\max_{j=1,\cdots,m\cap\hat{p}_h^j>t_j}\hat{p}_h^j$
19. //计算置信联合分布
20. 初始化 C 为 m×m 的全零矩阵
21. for i←1 to m do
22.　for k←1 to m do
23.　　$\mathbf{X}_{\tilde{y}=i,y^*=k}\leftarrow\varnothing$
24.　　for $\mathbf{x}_h\in\boldsymbol{X}_{\tilde{y}=i}$ do
25.　　if $y^*_{x_h}==$k then
26.　　$\mathbf{X}_{\tilde{y}=i,y^*=k}$.add($x_h$)
27.　　$\mathbf{C}_{\tilde{y}=i,y^*=k}\leftarrow|\mathbf{X}_{\tilde{y}=i,y^*=k}|$
28. //计算估计联合分布
29. 初始化 $\hat{\mathbf{Q}}$为 m×m 的全零矩阵
30. for i←1 to m do
31.　$\mathbf{C}_{\tilde{y}'=i}\leftarrow 0$
32.　for k←1 to m do
33.　　$\mathbf{C}_{\tilde{y}'=i}\leftarrow\mathbf{C}_{\tilde{y}'=i}+\mathbf{C}_{\tilde{y}=i,y^*=k}$
34.　for k←1 to m do
35.　　$\hat{\mathbf{Q}}_{\tilde{y}=i,y^*=k}\leftarrow\frac{1}{n}\cdot\frac{\mathbf{C}_{\tilde{y}=i,y^*=k}}{\mathbf{C}'_{\tilde{y}=i}}\cdot|\mathbf{X}_{\tilde{y}=i}|$
36. //选取每个类别的噪声样本
37. for i←1 to m do
38.　$d_{\tilde{y}=i}\leftarrow 0$
39.　$\hat{\mathbf{Q}}'_{\tilde{y}=i}\leftarrow 0$
40.　$\mathbf{X}'_{\tilde{y}=i}\leftarrow\varnothing$
41.　for k←1 to m do
42.　　if k≠i then
43.　　$\hat{\mathbf{Q}}'_{\tilde{y}=i}\leftarrow\hat{\mathbf{Q}}'_{\tilde{y}=i}+\hat{\mathbf{Q}}_{\tilde{y}=i,y^*=k}$
44.　　$d_{\tilde{y}=i}\leftarrow\lceil\hat{\mathbf{Q}}'_{\tilde{y}=i}\cdot n\rceil$
45.　将 $\mathbf{x}_h\in\boldsymbol{X}_{\tilde{y}=i}$样本按照$\hat{p}_h^i$升序排列；
46.　取前$d_{\tilde{y}=i}$个样本作为噪声样本$X'_{\tilde{y}=i}$。

示例：对于原标签类别为1的样本，$d_{\tilde{y}=1}=\lfloor 24\cdot(0.05+0)\rfloor=2$。由式(5)计算得到示例中3个类别相应的噪声数据个数分别为2，3，4个，对应的排序样本号分别为4，1；10，13，14；23，24，19，15。

步骤2　为了防止删除噪声后进一步加剧数据不平衡问题，我们通过设置样本数量阈值λ，基于步骤1对每个类别的噪声样本基于自置信度的排序进行如下3种判断操作。

(1)每个类别在删除所有噪声后样本数仍大于或等于样本数量阈值，即$|\boldsymbol{X}_{\tilde{y}=i}|-d_{\tilde{y}=i}\geqslant\lambda$，则直接全部删除。其中，$d_{\tilde{y}=i}$为人工标注为$i$的类别中需要删除的噪声样本个数。

示例：将λ设置为6。第三个类别样本数为11，减去该类别中所有要删除的噪声数4时，即11－4＝7＞6，满足样本数量阈值条件，可直接将噪声样本15，19，23，24删除。

(2)实际样本数少于样本数量阈值的类别，即$|\boldsymbol{X}_{\tilde{y}=i}|<\lambda$，直接收集噪声样本。

示例：第一类样本总数为5，经过比较，5＜6，样本总数不满足样本数量阈值6，直接将2个噪声样本1，4全部收集。

(3)删除所有噪声后样本数小于样本数量阈值但原样本数大于样本数量阈值，即$|\boldsymbol{X}_{\tilde{y}=i}|-d_{\tilde{y}=i}<\lambda$且$|\boldsymbol{X}_{\tilde{y}=i}|\geqslant\lambda$，则根据排序结果收集自置信度较低的样本直至样本总数等于样本数量阈值。

示例：第二个类别样本中，8－3＝5＜6，需要按照自置信度升序排列后，选取前1个噪声样本，即收集第10个样本。

步骤3　收集了噪声数据后，为了缓解类别之间的数据不平衡问题，不能单纯地将噪声数据抛弃，我们更希望通过人工标注纠正标签错误。但是由于真实数据类别复杂多样，虽然我们仅收集了样本数接近样本数量阈值的类别的噪声数据，但总量仍然十分庞大，直接全部进行人工标注不可行。因此，依据在主动学习[8]以保证分类精确度不减少的前提下尽量降低人工标注成本的思想，本文设计了一种基于 Self-Confidence[6]＋ reBvSB[8]的主动学习样例选择算法，其目的是从潜在错误标签集合中挑选出最有价值的样本供人工重新标注，从而增加低于样本数量阈值类别的样本数量。其具体过程如下。

首先，根据自置信度从集合中找到标签错误概率最大的样本$SC(\boldsymbol{x})$，如式(6)所示：

$$SC(\boldsymbol{x})=\arg\min(\hat{p}(\tilde{y}=i;\boldsymbol{x}\in\boldsymbol{X}_{\tilde{y}=i},\boldsymbol{\theta})) \tag{6}$$

其中，$\tilde{y}$表示人工打标的类别，$\hat{p}(\tilde{y}=i;x\in\boldsymbol{X}_{\tilde{y}=i},\boldsymbol{\theta})$ 表示被人工打标为 i 的样本打标正确的概率。

其次，采用 reBvSB[8] 算法选择分类超平面附近那些不确定性较大的边缘样例集，并去除样例中的孤立点，因为不确定性最大的样本的信息熵也越大，更加难以分类，且孤立点的周围大多仅存在几个样例，甚至没有，将它们加入数据集中对模型分类准确率提升不大[8]。其计算式如式(7)所示：

$$reBvSB(\boldsymbol{x})=\arg\min(p(y_{\text{Best}}\mid\boldsymbol{x}_i)-p(y_{\text{Second-Best}}\mid\boldsymbol{x}_i))-\{\boldsymbol{x}_i\mid\|\boldsymbol{x}_i-c_m\|>t\} \tag{7}$$

其中，$p(y_{\text{Best}}\mid x_i)$ 表示 x_i 属于最优类的不确定性概率，$p(y_{\text{Second-Best}}\mid x_i)$ 表示 x_i 属于次优类的不确定性概率，c_m 是聚类中心，t 是样本和聚类中心的距离差阈值。

接着，将上述两种方法相结合作为主动学习样例选择算法，并挑选出最具代表性的样本，$SC(x)$可以挑选出错误概率最大的样本，而 $reBvSB(x)$ 可以过滤掉孤立点并筛选出最具不确定性的样本。其计算式如式(8)所示：

$$S=\alpha*SC(\boldsymbol{x})+(1-\alpha)*reBvSB(\boldsymbol{x}) \tag{8}$$

其中，α 是 SC 和 reBvSB 两种方法结合的最佳比例值，该值由多次实验得到。

最后，专家对挑选的噪声进行重新标注后加入训练集供模型训练。当达到标注成本时，停止主动学习流程。这个过程大大降低了人工的标注成本并减少了样本中的噪声，更重要的是维持了数据集中各个类别间的平衡性。

3.2 主干网络训练模块

由于各个任务具有独特性及专业性，其中表达的语义信息也不同，而预训练模型是针对通用任务进行训练的，其学习到的是通用语义，应用于不同的任务种效果差强人意。因此，本文模型采用进一步预训练的方法来提高其学习能力。

我们采用与下游任务领域相同的数据集进行预训练并在同领域内的数据集进行微调。由于它们具有相似的数据分布和语义信息，因此该模型可以学习到更全面、准确的语义知识，并具有更好的分类能力。

由嵌入层、12 层 Transformer 编码层和分类器层 3 部分组成的主干网络如图 5 左侧所示，我们将主干网络作为教师模型，进一步进行模型压缩和加速。

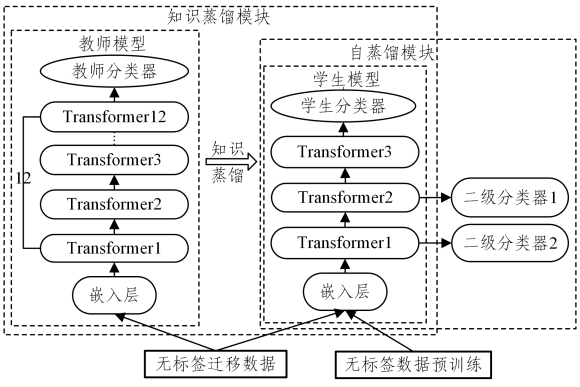

图 5 模型结构图

Fig. 5 Model structure graph

3.3 模型压缩和加速模块

主干网络(教师模型)具有 12 层 Transformer 结构，包含 110×10^6 的参数量，难以将其部署到有限的资源环境中。为了能成功部署模型，本模块使用知识蒸馏方法，将原本具有 12 层 Transformer 的大模型(教师模型)压缩成紧凑的具有 3 层 Transformer 的小模型，该小模型(学生模型)可以学习到大模型的知识以及泛化能力。我们选择学生模型层数为 3 的原因有两个：首先，针对模型的层数进行压缩时，要成倍压缩，即从教师模型中每两层或每三层映射到学生模型中的一层，如 BERT-PKD[17] 分别使用 2,4,6,8,10 教师层指导 1-5 学生层；其次，我们从 12 层可压缩的层数中选择 3 层，原因是在 FastBERT[12] 一文中，Liu 等在 12 个数据集上针对 1 层、3 层和 6 层的 DistilBERT[18] 进行了实验，表明在较小的精度损失下，3 层比 6 层的 DistilBERT 的速度快一倍，1 层 DistilBERT 模型的准确率相较于前两个模型最多降低了 12.78%。综合以上原因，我们选择的学生模型的层数为 3。此外，我们对学生模型进行预训练，以进一步提高模型的鲁棒性。

减小模型大小后，为了使模型具有自适应的能力，即灵活地调整计算步数并避免冗余计算，面对大规模数据量以及高效率需求时，我们运用多阶段自蒸馏方法，在学生模型的每一层 Transformer 后增加一个二级分类器，将学生分类器的知识迁移到二级分类器，在模型自身上进行多阶段的蒸馏学习。具体结构如图 5 右侧所示。二级分类器通过学习学生模型输出的高质量软目标(模型输出样本的预测概率)来进行训练。在训练时，冻结学生模型的参数。由于二级分类器之间相互独立，因此本文采用 KL 散度来衡量二级分类器的输出和软目标之间的差异，并用所有二级分类器的 KL 散度总和作为自蒸馏的损失，如式(9)所示：

$$Loss(p_{s_1},\cdots,p_{s_{L-1}},p_t)=\sum_{i=1}^{L-1}D_{\text{KL}}(p_{s_i},p_t) \tag{9}$$

其中，L 表示自蒸馏模型中 Transformer 的总层数，本模型中 $L=3$，p_{s_i} 是每一个二级分类器的输出分布，p_t 是三层模型的输出分布，D_{KL} 是每个二级分类器的损失。

自蒸馏过后，使用 BERT 模型预测样本类别。在模型分类器计算后会输出每个样本对应的概率分布，从中选出概率最大的值。若预测的类别和实际类别一致，则认为预测正确。为了减少模型的冗余计算，我们预先设置了一个 0～1 的预测阈值 μ 来判断样本是否已预测正确，并将 Transformer 的层数记为 l，$l=1,2$。在第 l 层 Transformer 中，样本会通过二级分类器输出预测概率，并选出概率分布中最大的值作为预测置信度 $Confidence_l$。一旦预测置信度超过预测阈值，即 $Confidence_l>\mu$，则认为该样本极有可能已经预测正确，将它提前在二级分类器中输出结果。当样本的预测置信度未超过预测阈值时，即 $Confidence_l\leqslant\mu$，我们令该样本在高层网络中输出，这样不仅减少了冗余计算，还增加了样本的自适应推理能力。在预测时，第 l 层 Transformer 中每条样本的预测置信度 $Confidence_l$ 的计算式如式(10)所示：

$$Confidence_l=\max\left(\frac{prob_l(i)}{\sum_{j=1}^{m}prob_l(j)}\right) \tag{10}$$

其中，m 表示类别总数，l 表示处于模型的第 l 层 Transfor-

mer，$prob_l(i)$表示第 l 层 Transformer 中每条样本的概率输出分布中第 i 个类别的概率，归一化后求得最大的概率并将其作为该样本最终的预测置信度$Confidence_l$。

不同的预测阈值设置影响着系统的预测能力及预测速度。例如，当预测阈值接近 1 时，绝大多数样本会在高层输出，模型的准确率会提高，同时速度也会下降。通过调节预测阈值和预测置信度，样本获得了自适应推理能力。在 4.5 节中将验证预测阈值、准确率和速度三者之间的关系。

4　实验

4.1　数据集

实验使用两个中文数据集对模型进行验证，包括电信数据集和书评数据集，涉及投诉服务和情感分析相关领域。

(1)电信数据集：由电信客服的投诉服务工单产生，包含 1014 个类别，其中每个类别表示电信领域的一种投诉工单类型，如基础业务、宽带和流量业务等。我们把数据集划分为训练集和测试集，分别占 76.81% 和 23.19%。其中，训练集是投诉工单的历史数据总和，含 530000 条样本，测试集是两个月的投诉服务工单总和，含 160000 条样本。

(2)书评数据集[1)]：包含豆瓣上关于书籍的评价，是一个用于句子分类任务的情感分析数据集[19]，共有近 40000 条样本，其中训练集包含 20001 条样本，测试集包含 10001 条样本，验证集包含 10001 条样本，分别占 50%，25%，25%。其中，类别 0 表示消极的书评，类别 1 表示积极的书评。

4.2　评价指标

本文使用了 3 种常见的评价指标来评价模型的性能，包括精度(*macro-P*)、召回率(*macro-R*)和准确率(*Accuracy*)[20]。计算方法如下：

$$macro\text{-}P=\frac{1}{n}\sum_{i=1}^{n}p_i \tag{11}$$

$$macro\text{-}R=\frac{1}{n}\sum_{i=1}^{n}r_i \tag{12}$$

$$Accuracy=\frac{True}{True+False} \tag{13}$$

其中，m 是类别的数目，p_i 是预测为类别 i 的样本中预测正确的样本所占比例，r_i 是真实类别为 i 的样本中预测正确的样本所占比例，$True$ 是预测正确的样本总量，$False$ 是预测错误的样本总量。

4.3　文本抗噪分析

在本节中，我们在电信数据集上对不同文本抗噪方法进行了分析，包含准确率、召回率和精度的对比；在 BERT 模型上分析了抗噪方法的有效性，同时在不同的样本数量阈值 λ 下分析了阈值抗噪的有效性。其中原始噪声数据指未进行处理的训练集；暴力去除噪声指直接删除置信学习找出的噪声；不同样本数量阈值的方法指对置信学习找出的噪声，根据样本数量阈值选择性地动态删除噪声样本。对于收集的噪声数据集，我们用主动学习找出最有价值的 5000 条样本重新进行人工标注并将它们加入训练集中。结果如表 1 所列。

表 1　抗噪方法性能分析

Table 1　Anti-noise method performance analysis

(单位：%)

方法	准确率	召回率	精度
原始噪声数据	48.06	18.22	28.04
暴力去除噪声	**49.76**	18.04	**29.12**
λ=200	49.16	19.93	27.79
λ=500	49.24	**20.02**	28.34

从表 1 中可以看到，与原始噪声数据相比，经过不同程度的噪声去除后，模型的准确率提升明显，在暴力去除噪声时提升了 1.7%，在样本数量阈值 λ 为 200 和 500 时，预测准确率也分别提高了 1.1% 和 1.18%。这是因为在去除噪声后，模型在训练过程中减少了噪声标签的干扰，使训练更加完善，进而提升了预测的准确率。但是暴力去除噪声的准确率比本文提出的阈值抗噪方法准确率更高，这是由于在数据集中，样本数大的类别占比较大，如表 1 所列，直接暴力去除噪声，虽然可以提升总样本的预测准确率，但其召回率比原始噪声数据的召回率低了 0.18%，更加验证了虽然暴力去噪的准确率有所提升，但暴力去噪方法只提升了模型对样本数较大类别的预测能力，并不能提升模型对每个类别样本的预测能力。相比之下，经过阈值抗噪后，模型的召回率比原始噪声数据分别增加了 1.71% 和 1.8%，这说明模型对每个类别的预测能力都有所提高。

通过对不同样本数量下类别的召回率进行对比，可将样本的数量划分成 7 个等级，如表 2 所列。

表 2　不同样本数量的类别召回率

Table 2　Category recall rates with different sample sizes

数量划分	类别数量/个	样本数量/个	原始数据/%	暴力抗噪/%	阈值抗噪/%
0～50	594	7571	9.59	9.40	**11.06**
50～100	75	5222	24.03	23.12	**26.95**
100～500	137	31500	33.91	34.76	**37.91**
500～1000	44	31501	45.89	46.96	**47.02**
1000～3000	29	48432	51.41	52.56	**53.13**
3000～10000	4	15928	46.43	**48.50**	48.36
10000 以上	2	26 985	61.30	**62.76**	62.45

首先，通过对比可以看到暴力抗噪以及阈值抗噪的召回率都比原始数据更高，说明抗噪可以提升模型效果。其次，对于表 2 中样本数在 3000 以下的类别，在经过阈值抗噪和主动学习重新标注后召回率得到提升，这表明了在某一类别样本数较少且不平衡时，相比单纯地删除所有噪声数据，选出最有价值的样本并重新进行人工标注后训练更能提升模型的准确率，此时模型不仅不受噪声数据的影响，而且干净数据的增加促使模型学习到了更全面的类别知识。例如样本数在 50～100 和 100～500 这两个量级的召回率分别提升了 2.92% 和 4%。而暴力去除噪声后样本数较少的类别的召回率反而下降，这也验证了暴力去除所有噪声的不合理性。例如样本数在 50～100 这个量级时召回率下降了 0.91%，这是由于暴力去除噪声可能会导致部分类别的数量接近于 0，进一步加剧不平衡问题，此时模型无法识别这些类别。最后，在类别具有较多样本的情况下，阈值和暴力抗噪的召回率均得到了提升，

1) https://github.com/yanchengxier/douban_book_review

说明在去除噪声样本后，减少了干扰，这些类别能够得到更好的训练，模型的准确率也随之提高。

4.4 主干网络分类方法

本节分别在两个数据集和不同的模型上验证全词掩盖和进一步预训练在模型预测准确率上的效果。其中涉及 3 个模型。

(1)BERT[4]：谷歌官方的中文预训练模型，对文本分类任务具有很好的效果。

(2)BERT-wwm[21]：哈工大在 BERT 的基础上将掩盖方式转变为全词掩盖。

(3)RoBERTa[22]：哈工大提供的基于 BERT 的动态全词掩盖的中文预训练模型。

其中，“FPT”表示在原模型的基础上进行了进一步的预训练。

表 3 全词掩盖和进一步预训练效果分析

Table 3 Full word masking and further pre-training effects analysis

(单位：%)

模型	电信数据集准确率	书评数据集准确率
BERT	49.16	88.54
BERT-wwm	49.47	88.63
RoBERTa	49.90	88.78
BERT-FPT	49.44	**89.08**
BERT-wwm-FPT	49.95	88.71
RoBERTa-FPT	**50.86**	88.98

从表 3 可以看出，在电信数据集中，经过全词掩盖的方法(BERT-wwm，RoBERTa)与原方法 BERT 相比均有较好的提升，准确率最高提升了 0.74%。进一步预训练的模型(BERT-FPT，BERT-wwm-FPT，RoBERTa-FPT)也在相应的原模型(BERT，BERT-wwm 和 RoBERTa)的基础上最高提升了 0.96%。但对于书评数据集，因其只包含两种类别，因此准确率提升不大。进一步预训练模型可以提升准确率的原因在于，经过预训练和全词掩盖后模型可以学到领域内的语义信息，并且基于全词掩盖的模型不再局限于学习单个字的语义，而是考虑到了中文文本数据中字与词的语义差异，能够学习到相关词语的语义。

4.5 模型压缩和加速功能

4.5.1 蒸馏预训练

我们在电信数据集上对以下 4 种模型作比较：

(1)ALBERT[9]：主要通过参数共享来减少模型的参数量，采用了目前较为流行的模型压缩技术。

(2)RoBERTa[22]：蒸馏预训练的基础模型。

(3)Rbt3-KD：通过知识蒸馏，将 RoBERTa 从原来的 12 层 Transformer 结构变成 3 层 Transformer 结构。

(4)Rbt3-KDP：在 Rbt3-KD 基础上进行蒸馏预训练后的模型。

以 RoBERTa 的训练速度为基准，对模型进行蒸馏预训练的实验结果如表 4 所列。从表中可看出，ALBERT 的模型最小，它采用参数共享和因式分解的方法减少了模型参数量，其训练速度并不会大幅度提升。虽然 Rbt3-KD 和 Rbt3-KDP 两个模型比 ALBERT 大 119MB，但速度却比其快 5 倍多，准确率最多提高了 0.71%。另外，与 RoBERTa 相比，Rbt3-KD 和 Rbt3-KDP 模型的大小压缩到了原来的 2/5，在训练速度上，因其模型结构简单，因此比 RoBERTa 快近 6 倍。在准确率方面，经过蒸馏预训练的 Rbt3-KD 模型相比 RoBERTa 模型性能只下降了约 2%，这是因为学生模型经过知识蒸馏后，很好地学习到了教师模型中包含的知识，使其具有与教师接近的性能。在现实生产中，小模型因为以上优点更容易部署与使用。

表 4 蒸馏预训练模型性能对比

Table 4 Performance comparison of distillation pre-trained models

模型	模型参数/($\times 10^6$)	训练速度/x	准确率/%
ALBERT	**44**	1.44	47.96
RoBERTa	395	1.00	**50.86**
Rbt3-KD	163	**8.14**	48.13
Rbt3-KDP	163	7.63	48.67

4.5.2 自适应预测

为了验证模型的自适应预测能力，我们将 Rbt3-KD 与 4 种基准方法在两个数据集上进行了对比分析，如表 5 所列，可以明显地看到 Rbt3-KD 模型在较小的准确率损失下，能获得 4～8 倍的加速。

(1)RoBERTa[22]：用于对比分析的基准模型。

(2)DistilBERT$_3$：3 层的 DistilBERT[18] 模型，原模型是由 Huggingface 提出的知识蒸馏模型。

(3)BERT-EMD$_3$：3 层的 BERT-EMD[23] 模型。BERT-EMD 模型是基于多层知识迁移的知识蒸馏方法，在准确率上超过了目前大部分知识蒸馏方法。

(4)FastBERT[12]：首次结合自蒸馏和自适应推理的模型。

表 5 自适应预测

Table 5 Adaptive forecast

模型	μ	电信数据集		书评数据集	
		准确率/%	速度/x	准确率/%	速度/x
RoBERTa	1.0	**50.86**	1.00	**88.98**	1.0
DistilBERT$_3$	1.0	—	—	81.17	4.01
BERT-EMD$_3$	1.0	48.61	2.87	85.46	**5.46**
FastBERT	0.2	42.08	**7.91**	80.84	4.90
	0.5	44.16	6.21	82.58	4.15
	0.8	48.23	3.26	85.40	3.08
Rbt3-KD	0.2	42.22	**7.91**	84.48	5.16
	0.5	45.21	6.87	84.85	4.66
	0.8	47.99	5.55	85.06	4.24

从表 5 可以看出，与 RoBERTa 相比，Rbt3-KD 可以在不同精度损失下加速 4～8 倍；与 DistilBERT$_3$ 相比，Rbt3-KD 在准确率和速度上都取得了更好的结果；与 BERT-EMD$_3$ 方法相比，Rbt3-KD 在准确率和速度上都有所下降。但在实际业务中，BERT-EMD$_3$ 的速度是固定的，无法应对需求及数据量变化的情况，相反 Rbt3-KD 由于具有速度自适应调节能力，会展现出更好的优势。另外，对于具有两阶段蒸馏框架的 TinyBERT[16]，我们没有将其放入自适应预测对比实验中，原因是在 BERT-EMD[23] 模型论文中 Li 等在 GLUE 上的 9 个任务上进行了 BERT-EMD 和 TinyBERT 的对比实验，结果表明 BERT-EMD$_4$ 和 BERT-EMD$_6$ 的准确率分别超过了 TinyBERT$_4$ 和 TinyBERT$_6$ 的准确率，因此 Rbt3-KD 模型只与

效果更好的 BERT-EMD 模型进行了对比实验。与 FastBERT 相比，当预测阈值 μ 较低时，Rbt3-KD 在准确率和速度上都取得了更好的结果，在准确率上领先于 FastBERT。当预测阈值 μ 较大时，Rbt3-KD 准确率虽低于 FastBERT，但速度远远超过它。这是因为预测阈值很高时，大多数样本在低层的网络满足训练完成的条件，而后大部分会在高层网络输出，并且 Rbt3-KD 是一个紧凑的 3 层 Transformer 结构，相较于 FastBERT 的 12 层结构，其在准确率上虽有所折损，但在实际场景下，我们更在意的是速度和模型大小，可以容忍精度上的部分损失。例如，在电信投诉服务中，模型会预测出最准确的 5 个类别，业务要求其中若有一个正确就认为该样本预测正确，并不考虑正确类别位于第几位，这说明精度损失是被允许的。

从表 5 中也可以进一步分析出准确率、预测阈值和处理相同样本数所需时间的关系。随着预测阈值的增大，准确率上升但速度下降，处理相同样本数所需时间会增加，这是因为预测阈值越大，过滤条件越严格，更多样本会在高层 Transformer 输出，准确率自然会提升，处理时间也呈增加趋势。这反应了冗余计算会导致处理样本所需时间的增加。面对不同需求以及数据量变化时，本模型可以调节适合的预测阈值，进而在客户关注度较高的需求层面上提高业务质量，这使本模型在工业界具有很大的吸引力。

结束语　本文提出了一种融合抗噪和双重蒸馏的文本分类方法。在数据抗噪模块，提出了阈值抗噪方法和一种新的主动学习样本选择策略算法。在主干网络模块，比较了不同模型的分类效果，选出了分类效果最好的模型。在模型压缩和加速模块，使用知识蒸馏和自蒸馏相结合的方式改进了模型架构，并提出了一种预测置信度和预测阈值的评估方法来实现自适应推理。本文在两个真实数据集中进行了充分的对比实验，验证了本模型具有良好的文本抗噪性能以及预测精度，可以很好地应用到各个领域，使业务办理更快捷，质量更优化。

本文方法也存在不足，例如，我们在数据降噪模块中增设了样本数量阈值来缓解降噪造成的数据不平衡问题，在一定程度上提升了模型的性能，但我们的样本数量阈值是通过多次实验得到合适的值，未来我们希望能够更进一步地研究自适应调整样本数量阈值的方法，减少设置样本数量阈值的复杂性。此外，本文方法牺牲了少量的精度来换取加倍的速度，希望之后能研究在保证精度的情况下提高速度的方法。我们也可以在小样本学习方面进一步展开研究，并将本文方法延伸至更深层次的预训练语言模型中。

参 考 文 献

[1] HAN X, ZHAO W, DING N, et al. Ptr: Prompt tuning with rules for text classification[J]. AI Open, 2022, 3, 182-192.

[2] MIKOLOV T, SUTSKEVER I, CHEN K, et al. Distributed representations of words and phrases and their compositionality [J]. Advances in Neural Information Processing Systems, 2013, 26, 3111-3119.

[3] VASWANI A, SHAZEER N, PARMAR N, et al. Attention is all you need[J]. Advances in Neural Information Processing Systems, 2017, 30, 6000-6010.

[4] DEVLIN J, CHANG M W, LEE K, et al. Bert: Pretraining of deep bidirectionaltransformers for language understanding[J]. arXiv: 1810. 04805, 2019.

[5] KOVALEVA O, ROMANOV A, ROGERS A, et al. Revealing the Dark Secrets of BERT[C]// Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP). Hongkong: Association for Computational Linguistics, 2019: 4365-4374.

[6] NORTHCUTT C, JIANG L, CHUANG I. Confident learning: Estimating uncertainty in dataset labels[J]. Journal of Artificial Intelligence Research, 2021, 70: 1373-1411.

[7] ZHANG H, CISSE M, DAUPHIN Y N, et al. mixup: Beyond empirical risk minimization[C] // International Conference on Learning Representations. Canada: OpenReview. net, 2018: 1-13.

[8] TIAN X X. An Improved Algorithm of Active Learning Based on Multiclass Classification [D]. Baoding: Hebei University, 2017.

[9] GORDON M, DUH K, ANDREWS N. Compressing BERT: Studying the Effects of Weight Pruning on Transfer Learning [C] // Proceedings of the 5thWorkshop on Representation Learning for NLP. On-line: Association for Computational Linguistics, 2020: 143-155.

[10] LAN Z, CHEN M, GOODMAN S, et al. ALBERT: A Lite BERT for Self-supervised Learning of La-nguage Representations[C] // International Conference on Learning Representations. Formerly Addis Ababa ETHIOPIA: 2019: 1-17.

[11] GOU J, YU B, MAYBANK S J, et al. Knowle-dge distillation: A survey[J]. International Journal of Computer Vision, 2021, 34: 1-31.

[12] LIU W, ZHOU P, ZHAO Z, et al. Fastbert: a self-distilling bert with adaptive inference time[C]// Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics. Online: Association for Computational Linguistics, 2020: 6035-6044.

[13] TANAKA D, IKAMI D, YAMASAKI T, et al. Joint optimization framework for learning withnois-y labels[C]// Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. Salt Lake City: IEEE, 2018: 5552-5560.

[14] LIN S, JI R, CHEN C, et al. ESPACE: Acceler-ating convolutional neural networks via eliminating spatial and channel redundancy[C]// Thirty-First AAAI Conference on Artificial Intelligence. San Francisco: AAAI Press, 2017: 1424-1430.

[15] ZAFRIR O, BOUDOUKH G, IZSAK P, et al. Q8bert: Quantized 8bit bert[C]// 2019 Fifth Workshop on Energy Efficient Machine Learning and Cognitive Computing-NeurIPS Edition (EMC2-NIPS). Vancouver: IEEE, 2019: 36-39.

[16] JIAO X Q, YIN Y C, SHANG L F, et al. TinyBERT: Distilling BERT for Natural Language Understanding[C] // Findings of

the Association for Computational Linguistics(EMNLP 2020) 2020:4163-4174.

[17] SUN S,CHENG Y,GAN Z,et al. Patient Knowledge Distillation for BERT Model Compression[C]// Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing(EMNLP-IJCNLP). Hongkong: Association for Computational Linguistics,2019:4323-4332.

[18] SANH V,DEBUTL,CHAUMONDJ,et al. DistilBERT,a distilled version of BERT:smaller,faster,cheaper and lighter[J]. arXiv:1910.01108,2019.

[19] QIU Y Y,LI H Z,LI S,et al. Revisiting correl-ations between intrinsic and extrinsic evaluations of word embeddings[C]// Chinese Computational Linguistics and Natural Language Processing Based on Naturally Annotated Big Data. Cham: Springer, 2018. 209-221.

[20] SCHÜTZE H,MANNING C D,RAGHAVAN P. Introduction to information retrieval[M]. Cambridge: Cambridge University Press,2008.

[21] CUI Y,CHE W,LIU T,et al. Pre-training with whole word masking for chinese bert[J]. IEEE/ACM Transactions on Audio, Speech,and Language Processing,2021,29:3504-3514.

[22] LIU Y,OTT M,GOYAL N,et al. RoBERTa:A Robustly Optimized BERT Pretraining Approach[J]. arXiv: 1907.11692, 2019.

[23] LI J,LIU X,ZHAO H,et al. BERT-EMD:Many-to-Many Layer Mapping for BERT Compression with Earth Mover's Distance [C]// Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing(EMNLP). Online: Association for Computational Linguistics,2020:3009-3018.

GUO Wei, born in 2001, postgraduate, is a member of China Computer Federation. Her main research interest is natural language processing.

HOU Chenyu, born in 1994, Ph.D, lecturer, Ph.D supervisor, is a member of China Computer Federation. His main research interests include data mining and so on.

(责任编辑:何杨)