计算机系统应用 ISSN 1003-3254, CODEN CSAOBN
Computer Systems & Applications,2021,30(7):50−56 [doi: 10.15888/j.cnki.csa.008031]

E-mail: csa@iscas.ac.cn
http://www.c-s-a.org.cn
Tel: +86-10-62661041

联合知识的融合训练模型①

王永鹏[1], 周晓磊[1,2], 马慧敏[2], 曹吉龙[2]

[1](中国科学院 沈阳计算技术研究所, 沈阳 110168)
[2](中国科学院大学, 北京 100049)
[3](东软集团股份有限公司, 沈阳 110179)
[4](中国医科大学附属第四医院, 沈阳 110032)
通讯作者: 周晓磊, E-mail: zhouxl@sict.ac.cn

摘　要: 在互联网医疗领域, 智能 AI 分科室是一个很关键的环节, 就是根据患者病情描述、疾病特征、药品等信息将患者分配到匹配的科室, 可以利用深层双向 Transformer 结构的 BERT 预训练语言模型增强字的语义, 但是患者病情文本描述具有信息稀疏的特点, 不利于 BERT 的充分学习其中特征. 本文提出了一种 DNNBERT 模型. 是一种融合知识的联合训练模型, DNNBERT 结合了神经网络 (DNN) 和 Transformer 模型的优势, 能从文本中学习到更多的语义. 实验证明 DNNBERT 的计算时间相比 BERT-large 速度提升 1.7 倍, 并且准确率比 ALBERT 的 *F*1 值提高了 0.12, 比 TextCNN 提高了 0.17, 本文的工作将为特征稀疏学习提供新思路, 也将为基于深度 Transformer 的模型应用于生产提供新的思路.

关键词: 知识融合; 医疗短文本; BERT 模型; 联合训练; 文本分类

引用格式: 王永鹏,周晓磊,马慧敏,曹吉龙.联合知识的融合训练模型.计算机系统应用,2021,30(7):50–56. http://www.c-s-a.org.cn/1003-3254/8031.html

Ensemble Training Model Integrating Knowledge

WANG Yong-Peng[1], ZHOU Xiao-Lei[1,2], MA Hui-Min[2], CAO Ji-Long[2]

[1](Shenyang Institute of Computing Technology, Chinese Academy of Sciences, Shenyang 110168, China)
[2](University of Chinese Academy of Sciences, Beijing 100049, China)
[3](Neusoft Group Co. Ltd., Shenyang 110179, China)
[4](The Fourth Affiliated Hospital of China Medical University, Shenyang 110032, China)

Abstract: In the field of Internet-based medical treatment, AI-based triage represents a key link, which allocates patients to departments according to conditions, disease attributes, medications, etc. We can adopt the BERT with a deep bi-directional Transformer structure for language model pre-training to enhance the word semantics; however, the text description of patients' conditions offers sparse information, which is not conducive to the full learning of characteristics by BERT. This paper presents DNNBERT, a joint training model integrating knowledge. Combining the advantages of Deep Neural Network (DNN) and the Transformer model, DNNBERT can learn more semantics from text. The experimental results prove that the computing time of DNNBERT is 1.7 times shorter than that of BERT-large; the accuracy rate of DNNBERT is 0.12 higher than the *F*1 value of ALBERT and 0.17 higher than that of TextCNN. This paper will provide a new idea for sparse feature learning and the applications of deep Transformer-based models to production.

Key words: knowledge fusion; medical short text; BERT model; joint training; text classification

① 基金项目: 国家水体污染控制与治理科技重大专项 (2012ZX07505)
Foundation item: Major Special Projects of Science and Technology for National Water Pollution Control and Treatment of China (2012ZX07505)
收稿时间: 2020-11-04; 修改时间: 2020-12-12, 2020-12-18, 2020-12-25; 采用时间: 2021-01-06; csa 在线出版时间: 2021-06-30

线下医院排队时间长, 看病慢已经是每个患者心中的痛点. 现在虽然几乎每个医院都有线上预约挂号, 线下看病, 或者线上直接问诊等功能但是还需要患者根据自己的病情特征自己选择科室, 进而找到医生, 但是这种方式显示没有将互联网医院数据得到充分的利用. 因而利用深度学习的技术解决人们快速的线上就诊成为急切的问题.

医患 IM 页作为最大的流量和服务入口, 其中聊天文本信息覆盖了疾病名称、症状、药品名称等多种命名实体, 所以准确的刻画 Query 和 Context 之间的深度语义相关性至关重要, 还有不同患者在描述症状上字面相差巨大, 比如感冒和头痛、鼻子堵塞、流鼻涕、发烧、身体发冷等词之间的字面意思相差非常大, 但是他们却是和呼吸内科是语义相关的. 在这种场景下用户希望在不同场景下通过输入文字来获取自己想要的服务, 这就需要一个强大的模型.

这其中有 4 个痛点: 第 1 个是患者描述病情信息特征稀疏问题, 模型很难学习到准确的语义信息. 第 2 个是基于深度 Transformer 的模型虽然效果好, 但是推理速度慢, 很难应用到实时的生产环境. 第 3 个是文本医患聊天中命名实体信息多而复杂. 第 4 个是不同患者描述字面信息差别大. 为了解决这几个痛点问题, 本文提出了自己设计搭建的 DNNBERT 算法模型.

这样的设计灵感来源是用户的信息特征可能随时都会横向增加, 来弥补数据稀疏问题, 例如实现智能分诊仅仅依靠患者简单粗糙的病情描述是比较困难的, 还需要考虑患者年龄, 性别, 历史疾病等横向的信息, 因此设计出 DNNBERT, 其中 DNN 部分可以处理用户附加特征, 基于 BERT 改进的 ALBERT[1] 模型可以充分发挥其强大的语义理解能力, 所以 DNNBERT, 不仅解决了文本向量信息稀疏的问题, 还解决了 BERT 模型训练速度慢的问题, 将深层 Transformer 模型应用于生产提供了新思路.

自然语言处理中文本分类问题, 最重要的就是文本表示, 就是如何提取出文本中原本的语义信息, 最早期的二分类学习模型采用 one-hot 编码, 仅单纯地将文本表示为算法可处理的结构化向量, 参数量也会非常大. CNN 采用共享卷积核方式, 优化了参数量, 并且深层次的网络抽取信息更加丰富, 表达效果也更好, 但 CNN 的输入和输出都是相互独立的, 并没有考虑到文本中字前后的序列问题. RNN 引入了“记忆”的概念, 引入了时间序列模型, 输出依赖于输入和“记忆”, RNN 善于学习顺序建模任务但在提取特征的时候不能以并行的方式进行. 双向 RNN 采用双向编码特征拼接的方式, 让当前的输出不仅依赖于输入和之前的序列元素, 还依赖于之后的序列元素. 深层双向 RNN 在双向 RNN 的基础上改进, 在每个时间点设定多层网络结构. LSTM[2] 与 RNN 的基本结构相似, 区别是它提出了一种“记忆细胞”的概念, 该记的信息会一直传递, 不该记得会被“门”截断, 解决 RNN 远距离的信息丧失的问题. GRU[3] 是 LSTM 的变种, 将忘记门和输入门合成了一个单一的更新门, 混合了细胞状态和隐藏状态和其他一些改动. 它比标准 LSTM 更简单. Word2Vec[4] 将文本的表示通过词向量的表示方法, 把文本数据从高纬度稀疏的神经网络难处理的方式, 变成了类似图像、语言的连续稠密数据. Google 的词向量文章中涉及的两个模型 CBOW (上下文来预测当前词) 和 Skip-gram (当前词预测上下文). ELMo[5] 来自于语言模型的词向量表示, 也是利用了深度上下文单词表征, 该模型的好处是: (1) 能够处理单词用法中的复杂特性 (比如句法和语义); (2) 这些用法在不同的语言上下文中如何变化 (比如为词的多义性建模). GPT[6] 这种模型之所以效果好是因为在每个新单词产生后, 该单词就被添加在之前生成的单词序列后面, 这个序列会成为模型下一步的新输入. 这种机制叫做自回归 (auto-regression), 同时也是令 RNN 模型效果拔群的重要思想. 而 BERT[7] 虽然没有使用自回归机制, 但 BERT 获得了结合单词前后的上下文信息的能力, 从而取得了更好的效果. XLNet[8] 使用了自回归, 并且引入了一种能够同时兼顾前后的上下文信息的方法. ALBERT 相比 BERT-LARGE, 在推理速度上有了巨大的改进, 具体的创新部分有 3 个: (1) 将 embedding 的参数进行了因式分解; (2) 跨层的参数共享; (3) 抛弃了原来的 NSP 任务, 现在使用 SOP 任务. 上面这些模型其实在文本分类效果上都具有巨大的意义的提高, 但是同时他们都具有超大的参数量, 从而导致在生产应用上显得有些笨重. 本文提出的 DNNBERT 就是为了解决上述问题.

1 相关研究工作

本文的研究主要致力于自然语言处理中语义理解方向, 研究内容为深层 Transformer 网络结构. 与 RNN 相比, Transformer 语义特征提取能力更强, 具备长距离

特征捕获能力, 基于注意力的深层 Transformer 模型[9]的成功启发了大量后续工作. 类似于智能分诊这样的短文本 (信息稀疏) 分类场景下, 学术界也出现了很多优秀的研究成果, 例如融合更多外部知识的百度 ERNIE[10], K-BERT[11]. 优化预训练目标的 ER-NIE2.0[12], RoBERTa[13], SpanBERT[14], StructBERT[15]等. 优化模型结构或者训练方式的 ALBERT. 关于预训练模型的各种后续工作, 可以参考复旦大学邱锡鹏老师最近的综述[16].

基于预训练好的 BERT 模型可以支持多种下游 NLP 任务. BERT 在下游任务中的应用主要有两种方式: 即 Feature-based 和 Finetune-based. 其中 Feature-based 方法将 BERT 作为文本编码器获取文本表示向量, 从而完成文本相似度计算、向量召回等任务. 而 Finetune-based 方法是在预训练模型的基础上, 使用具体任务的部分训练数据进行训练, 从而针对性地修正预训练阶段获得的网络参数. 该方法更为主流, 在大多数任务上效果也更好.

由于 BERT 家族类模型在 NLP 任务上的显著优势, 一些研究工作开始将 BERT 应用于复杂场景的文本分类任务上. 清华大学 Qiao 等人[17] 详细对比了 Feature-based 和 Finetune-based 两种应用方式在文本段落排序 (passage ranking) 中的效果, 分析说明了 BERT 如何在它的转换层中的查询文档词条之间分配注意力, 以及如何在释义词条之间选择语义匹配. 滑铁卢大学 Jimmy Lin 团队[18] 针对分类排序任务提出了基于 Pointwise 和 Pairwise 训练目标的 MonoBERT 和 DuoBERT 模型. 此外, 该团队提出融合基于 BERT 的 Query-Doc 相关性和 Query-Sentence 相关性来优化分类结果排序任务的方案. 为了优化文本分类性能和效果, Bing 广告团队提出一种双塔结构的 TwinBERT[19] 分别编码 Query 和 Context 文本. 2019 年 10 月, Google 在其官方博客中介绍了 BERT 在 Google 搜索排序场景的应用, BERT 强大的语义理解能力改善了约 10% 的搜索结果, 除了英文网页, Google 也正在基于 BERT 优化其他语言的搜索结果.

本文在这些前辈的思想和基础上, 针对医患聊天这样的特定场景, 提出了双塔结构的知识融合模型 DNNBERT, 它的创新点主要有: 可以不断的横向实现知识融合, 在预训练阶段引入患者大量结构化信息, 例如性别, 年龄, 历史疾病等标签信息, 弥补 Query 文本信息的不足, 强化语义匹配效果. 极大提高了模型运行的速度, 不同于 BERT-LARGE, 本文在融合模型时选择了 ALBERT 模型和 DNN 进行融合, DNNBERT 可以实现相对于 BERT-LARGE, 速度提升 1.7 倍.

2 模型介绍

随着智能化的发展, 越来越多的患者问诊就医只想通过一句话描述来快速的匹配到科室就医甚至直接获取自己想要的医生, 药品, 病情诊断等信息. 但是用户的文本描述是多种多样的, 既有药品名称, 疾病名称, 病情症状名称, 甚至同一种疾病治疗还有不同的药品名, 不同的症状描述, 不同的患者有不同的病情描述. 还有患者的基本特征, 包括年龄, 性别, 历史疾病, 最近浏览科室、最近浏览过的医生、最近问诊过的科室最近浏览过的医生、最近一次购药明细、最近病例描述等特征对于分科室也是至关重要的. 要将这么多维的信息来分类匹配到具体的科室, 准确的刻画数据就变得非常重要. 本文提出 DNNBERT 模型, 在特征表示上采用固定长度患者特征拼接患者文本描述的方式进行 Embedding. 模型网络设计分为两个网络, DNN 网络负责用户特征的信息抽取, Transformer 网络来学习文本描述. 将学习到的结果拼接链接到一个池化层, 再用 Softmax 进行分类, 模型架构图如图 1 所示.

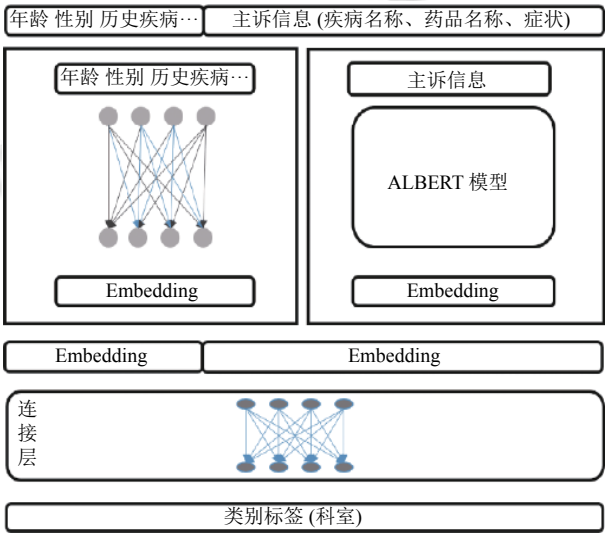

图 1 DNNBERT 模型结构图

从图 1 中可以看到, 自上而下, 从左到右, DNNBERT 模型主要由神经网络和 ALBERT 模型两部分排列进行处理, 类似于双塔并列, 所以称为双塔结构, 设计的初衷是并列的两个模型根据自身模型的特点来处理不

同特征的文本, DNN 模型对于文本的学习能力强, 像性别, 年龄这样的特征数据没有太多的深度语义, 所以采用 DNN 是不错的选择, 但是患者病情描述文本语义就比较复杂, 需要基于注意力的 Transformer 来学习其中的语义, 所以采用当前效果较好, 速度较快的 ALBERT, DNN 和 ALBERT 两者根据自身的优势来分别对文本进行学习, 最后将 DNN 和 ALBERT 模型学习的结果进行拼接, 从而 DNNBERT 输出的结果是模型结合了两个模型分别学习处理过得高度抽象语义信息, 实现了 DNN 和 Transformer 间知识补充和融合. 下面分别阐述模型主要模块的具体实现细节.

2.1 双塔-DNN 模型设计

图 1 中, 从上至下, 第 2 层左侧的神经网络模型主要学习患者相关基本信息, 例如: 性别 (两种可能 2 位进行编码)、年龄 (11 种可能 11 位编码, 1 至 10 岁为一类, 11 至 20 岁为一类, 一次类推)、最近一次访问的科室 (15 种可能 15 位编码)、最近一次问诊的医生所属科室 (15 种可能 15 位编码)、对最近一次问诊的医生的评价 (4 种可能 4 位编码), 然后进行最大最小归一化, 拼接成 feature 总共 64 维, 作为模型输入, 然后连接 DNN 层 (共两个隐藏层: 第一个隐藏层 64 个神经元, 该层使用 ReLU 激活函数; 第二个隐藏层 128 个神经元), 第三层输出层为 16 个神经元输出层为 16 维的特征向量, 作为 DNN 特征提取的结果将和 ALBERT 模型的拼接一起作为连接层的输入.

2.2 双塔-ALBERT 模型

图 1 中, 从上至下, 第 2 层右侧的模型为 ALBERT 模型, 该模型参数量小, 速度快, 用于学习患者病情描述文本的语义.

ALBERT 采用了两种参数简化技术, 消除了缩放预先训练模型的主要障碍. 第一种是因子化嵌入参数化. 通过将大的词汇嵌入矩阵分解为两个小矩阵, 我们将隐藏层的大小与词汇嵌入的大小分开. 这种分离使得在不显著增加词汇表嵌入的参数大小的情况下更容易增加隐藏的大小. 第二种技术是跨层参数共享. 该技术可防止参数随网络深度的增加而增大. 这两种技术在不严重影响性能的情况下显著减少了 BERT 的参数数目, 从而提高了参数效率. 参数简约技术也可以作为一种正则化的形式来稳定训练并有助于泛化. 为了进一步提高 ALBERT 的性能, 我们还引入了一种自监督的句子顺序预测损失 (SOP). SOP 主要关注句子间的连贯性, 旨在解决原 BERT 中提出的下一句预测 (NSP) 损失的无效性.

2.2.1 BERT 模型

BERT 根据名称就可以知道, 其核心是基于 Transformer 结构的, Transformer 是一个端到端训练的网络结构, 即输入和输出均为一个序列, Transformer 是一个由 Encoder 和 Decoder 组成的网络结构, 其中最重要的是在 RNN 的基础上加入了注意力机制. Encoder 由 6 个相同的 layer 组成. 每个 layer 由两个 sublayer 组成, 分别是 multiHead self-attention mechanism 和 fully connected feedforward network. 其中每个 sublayer 都加了 residual connection 和 normalization, multiHead attention 则是通过 h 个不同的线性变换对 Q, K, V 进行投影, 最后将不同的 attention 结果拼接起来.

Transformer 中最重要的两个结构是 Encoder 和 Decoder 结构, 我们先来看看 Encoder 结构的组成部分, 如图 2 所示.

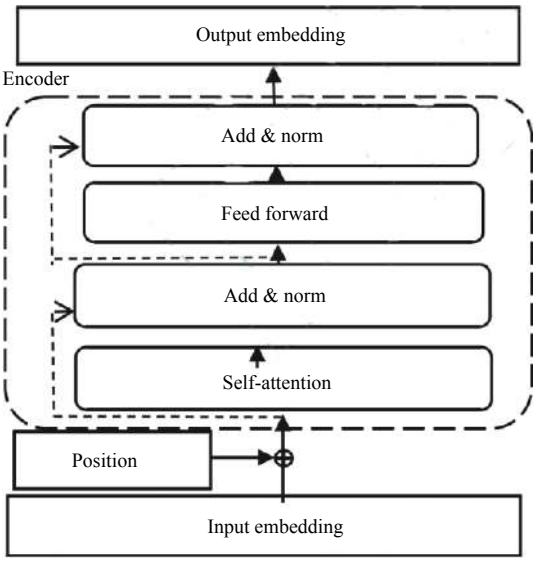

图 2 Encoder 内部结构示意图

如图 2 所示, Transformer 中没有采用传统的 CNN 和 RNN, 整个网络结构主要是由 self-Attenion 和 feed forward neural network 组成. 在经典的 BERT 模型中, Transformer 结构主要由 6 层的 Encoded 和 6 层的 Decoder 组成.

输入是文本的 Embedding 表示, 并且在输入的 Embedding 上加入了位置信息, 将结果输入自注意力层进行权重计算学习, 将计算的输出和自注意力的输入相加和归一化操作, 采用 Attention 机制的原因是考

虑到 RNN(或者 LSTM, GRU 等) 的计算限制为是顺序的, 也就是说 RNN 相关算法只能从左向右依次计算或者从右向左依次计算, 这种机制带来了两个问题:

时间片 t 的计算依赖 t–1 时刻的计算结果, 这样限制了模型的并行能力;

顺序计算的过程中信息会丢失, 尽管 LSTM 等门机制的结构一定程度上缓解了长期依赖的问题, 但是对于特别长期的依赖现象, LSTM 依旧无能为力.

Transformer 的提出解决了上面两个问题, 首先它使用了 Attention 机制, 将序列中的任意两个位置之间的距离是缩小为一个常量; 其次它不是类似 RNN 的顺序结构, 因此具有更好的并行性.

再将注意力输出的结果接入前馈神经网络, 进行学习输出 Embedding.

2.2.2 Attention 的计算过程

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

式中, d_k 的含义是每个字的 Q 向量的维度, 将 Q 和 K 向量的乘积结果进行 *Softmax* 归一化和 Value 向量相乘, 这样计算结束之后，就得到了每个字的注意力权重, 而后将上式中的每个结果进行累加, 就得到每个字的表示, 此时每个字已经融入了句子中其他字的信息进去, 是一个表达能力非常强的向量表示.

多头注意力 (multi-head attention) 是利用多个查询组成的矩阵 Q, 来平行地计算从输入信息中选取多个信息. 每个注意力关注输入信息的不同部分, 然后再进行拼接, 这样就可以从指代消解, 位置等不同维度来综合的识别语义信息, 计算公式如下:

$$MultiHead(Q,K,V)=Concat(head_1,head_2,\cdots,head_k)W^O$$

$$head_i=Attention\left(QW_i^Q,KW_i^k,VW_i^V\right)$$

在 BERT 模型中, 多头数量是一个超参数, 可以进行配置.

2.3 全连接层

如图 1 所示, 其中全连接层部分主要思路是将上侧 DNN 和 ALBERT 模型的输出拼接, 一方面进行联合训练学习, 抽象提取两部分模型的组合特征; 另一方面将上侧两部分特征进行压缩, 输出为 16 维特征, 目的是减少特征维度, 提高线上模型的运行性能. 该部分 DNN 模型共 3 个隐藏层: 第 1 个隐藏层 256 个神经元 (DNN 和 ALBERT 模型拼接结果), 该层使用 ReLU 激活函数; 第 2 个隐藏层 128 个神经元, 激活函数为 ReLU; 第 3 个隐藏层 64 个神经元, 激活函数为 *Softmax*, 输出层为 16 维分类的结果.

3 实验分析

3.1 实验数据

实验数据中主要列举了科室类以及每个科室类别对应的数据条数, 以及数据的分割, 分为训练集、验证集、测试集 3 类. 实验数据分布如表 1.

表 1 实验数据分布表

标签	内科	精神心理科	妇产科	儿科	眼科	皮肤性病科	耳鼻喉头颈外科	口腔科	骨科	中医科	其他科	外科	男科	总数	训练集	验证集	测试集
条数	136914	8048	29730	27882	16025	56471	16510	5192	17051	28391	326094	27976	35763	738234	590587	73823	73823

本文实验使用的语料库来自京东互联网医院的患者就诊数据, 此次实验数据总条数约为 73 万条, 科室分类数为 15 个类别, 分别为内科, 外科, 妇产科, 儿科, 精神心理科, 肿瘤科, 中医科, 骨科, 眼科, 皮肤性病科, 耳鼻喉头颈外科, 口腔科, 男科, 整形美容科, 其他科.本次实验将总数按照 8:2 的比例分为训练集, 测试集.对模型迭代 10 000 次进行训练学习.并且在模型学习完毕后, 将数据随机打散分为五份数据, 分别在 A、B、C、D、E 五份样本数据的基础上和 TextCNN[20]、TextCNN_Att、FastText[21]、 BERT-LARGE、ALBERT 等 BERT 家族的模型进行了对比, 实验证明, 基于联合训练的模型 DNNBERT, 要比其他模型更容易学习到语义信息, 总体准确性更好.

3.2 评价指标

在进行模型评价的时候, 作者采用的是 $F1$ Score, 因为它更加能反映出模型的健稳性, 它被定义为模型精度和召回率的调和平均值, 因此, 如果你想在精确度和召回率之间寻求平衡, $F1$ Score 是一个更好的衡量标准.

在分类问题中, 我们都会用到混淆矩阵[22] 来计算和评估模型的性能, 如表 2 所示.

1) 精确度. 精确度给出了所有预测为正的结果中正确识别为正例的分数:

$$P=\frac{TP}{TP+FP}$$

2) 召回率. 召回率给出模型正确识别为正例的分数:

$$R=\frac{TP}{TP+FN}$$

3) $F1$, 可以理解为模型在召回和精准率之间做了调和平均:

$$F1=\frac{2*P*R}{P+R}$$

因此, 为了想在精确度和召回率之间寻求平衡的评价指标, $F1$ 分数是一个更好的衡量标准.

表 2 分类结果的混淆矩阵

	预测为正	预测为反例
真实为正例	*TP*	*FN*
真实为反例	*FP*	*TN*

3.3 实验过程

本次实验为了对比 DNNBERT 模型的优劣性, 分别和 TextCNN, TextCNN_Att, ALBERT, FastText, BERT-LARGE, 几种模型进行了对比, 表 3 是 DNNBERT 模型的参数设置.

表 3 DNNBERT 模型参数设置

参数名称	参数值
hidden_dropout_prob	0.1
hidden_size	768
num_hidden_layers	12
num_attention_heads	12
intermediate_size	3072
hidden_act	gelu
embedding_size	128
learning_rate	1e–5

表 3 中 hidden_ dropout_prob 表示隐层 dropout 率, hidden_size 表示隐藏层神经元数, num_hidden_layers 表示 Transformer encoder 中的隐藏层数, num_attenti on_heads 表示 multi-head attention 的 head 数 intermediate_size 表示 ALBERT encoder 的中间隐层神经元数 (例如 feed-forward layer), hidd - en_act: 隐藏层所采用的激活函数, embedding _ize 表示输入向量的大小, 短补长切.

实验采用的机器参数 Linux 系统, CPU 核数为 8 核, 内存大小为 40 GB, 一张 GPU 卡, batchSize 表示训练时每批数据的数量, 每批次数据量为 1000 条, epoch 表示迭代训练的次数, 训练 10 000 次, learning_ rate 学习率表示模型学习的准确率.

3.4 实验结果

本文先后做了 5 组实验, 分别为 A 组、B 组、C 组、D 组、E 组. 在进行模型对比实验时, DNNBERT, TextCNN_Att, ALBERT, TextCNN, FastText 等 5 个模型的参数不变, A, B, C, D, E 五组的数据也不变, 评价指标主要采用了综合精确率和召回率的 $F1$ 值进行对模型进度评估, 在相同实验环境下, 5 组实验 $F1$ Score 精度对比结果如表 4 所示.

图 3 是对应的表 4 数据的可视化, 即每个模型在每组实验数据集上的 $F1$ Score 的可视化展示.

表 4 模型精度对比表

模型	A	B	C	D	E
TextCNN_Att	0.5	0.8	0.76	0.8	0.78
DNNBERT	0.67	0.73	0.8	0.85	0.9
ALBERT	0.63	0.75	0.84	0.71	0.78
TextCNN	0.7	0.73	0.73	0.77	0.73
FastText	0.65	0.76	0.67	0.76	0.67

我们选择选择的模型评价标准为 $F1$ Score, 横坐标 1,2,3,4,5 分别表示 5 组实验, 纵坐标分别表示模型的 $F1$ 值, 范围从 0 到 1, 从图 3 中可以得出以下结论.

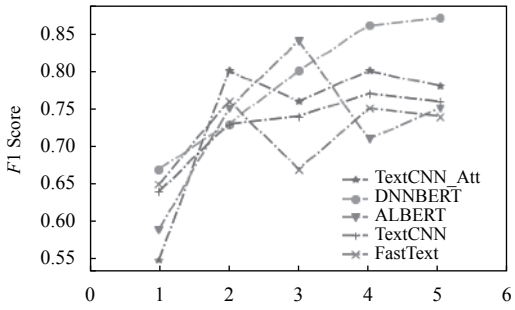

图 3 模型精度对比图

(1) 模型的精度随着训练数据量的增加而正比提高, 在第一次实验中, 可以观察到模型的精度层次不齐, 随着后面多次实验, 精度逐渐趋于稳定.

(2) DNNBERT 模型随着训练次数的增加, 效果相比其他对比模型, 精度领先于其他模型.

(3) 在经过 5 次实验之后, DNNBERT 模型的精度最高, 其次是 TextCNN_Att 模型, 即 DNNBERT 相比 TextCNN_Att, 精度提高了 0.12, 分数最低的是 FastText 模型, $F1$ 值为 0.67, DNNBERT 相比 FastText, 精度提高了 0.23.

综上, 我们可以得出结论, 本文提出的联合知识的融合训练模型, 在处理短文本分类场景时, 十分具有优越性, 同时也说明了集成训练模型的优越性, 成为了只能分诊场景最优的解决方案.

4 结论与展望

本文在解决中文短文本分类的问题中, 提出了一种基于 ALBERT 模型和 DNN 网络的双塔模型的知识

融合训练模型 DNNBERT, 并与 TextCNN, ALBERT, TextCNN_Att, FastText 模型进行对比, 先后做了 A 组、B 组、C 组、D 组、E 组 5 组实验, 对比实验时的测试数据量是总数据的 1/5. 5 组实验测试数据总数为 73 823, 平均分为 5 组, 每组 14 764 条. 实验结果表明, 融合知识的联合训练模型 DNNBERT 模型在科室分类中效果好于 CNN (TextCNN,TextCNN_Att, FastText) 和 ALBERT. 下一步将对模型中文本编码的长度与神经网络隐藏层的个数和预测结果间的关系进行研究, 来找出最优的模型参数, 并且也会让模型不断的融合更多的外部知识, 提高模型的泛化性. 除了模型的精度提升之外, 还会在模型的速度方面去提升, 后期希望引入蒸馏, 剪枝等技术将自己的模型不但从精度方面提升到最强, 也希望训练的速度足够快, 从而可以适用于工业生产.

致谢

特别感谢京东健康提供实验所需相关的数据, 感谢公司领导的大力支持.

参考文献

1 Lan ZZ, Chen MD, Goodman S, *et al*. ALBERT: A lite BERT for self-supervised learning of language representations. arXiv preprint arXiv: 1909.11942, 2019.

2 Guo T, Lin T. Multi-variable LSTM neural network for autoregressive exogenous model. arXiv preprint arXiv: 1806.06384, 2018.

3 Cho K, Van Merriënboer B, Gulcehre C, *et al*. Learning phrase representations using RNN encoder-decoder for statistical machine translation. arXiv preprint arXiv: 1406.1078, 2014.

4 Mikolov T, Chen K, Corrado G, *et al*. Efficient estimation of word representations in vector space. Proceedings of the 1st International Conference on Learning Representations. arXiv preprint arXiv: 1301.3781, 2013.

5 Peters ME, Neumann M, Iyyer M, *et al*. Deep contextualized word representations. Proceedings of 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies. New Orleans, LA, USA. 2018. 2227–2237.

6 Radford A, Narasimhan K, Salimans T, *et al*. Improving language understanding by generative pre-training. https://s3-us-west-2.amazonaws.com/openai-assets/research-covers/language-unsupervised/language_understanding_paper.pdf. [2020-10].

7 Devlin J, Chang MW, Lee K, *et al*. BERT: Pre-training of deep bidirectional transformers for language understanding. Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies. Minneapolis, MN, USA. 2019. 4171–4186.

8 Yang Z, Dai Z, Yang Y, *et al*. XLNet: Generalized autoregressive pretraining for language understanding. 2019.

9 Vaswani A, Shazeer N, Parmar N. *et al*. Attention is all you need. Proceedings of the 31st Conference on Neural Information Processing Systems. Long Beach, CA, USA. 2017. 5998–6008.

10 Zhang ZY, Han X, Liu ZY, *et al*. ERNIE: Enhanced language representation with informative entities. Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics. Florence, Italy. 2019. 1441–1451.

11 Liu WJ, Zhou P, Zhao Z, *et al*. K-BERT: Enabling language representation with knowledge graph. Proceedings of the AAAI Conference on Artificial Intelligence. New York, NY, USA. 2020. 2901–2908.

12 Sun Y, Wang S, Li Y, *et al*. ERNIE 2.0: A continual pre-training framework for language understanding. arXiv preprint arXiv: 1907.12412, 2019.

13 Liu Y, Ott M, Goyal N, *et al*. Roberta: A robustly optimized BERT pretraining approach. arXiv preprint arXiv: 1907.11692, 2019.

14 Joshi M, Chen D, Liu Y, *et al*. Spanbert: Improving pre-training by representing and predicting spans. Transactions of the Association for Computational Linguistics. 2020, 8: 64–77.

15 Wang W, Bi B, Yan M, *et al*. StructBERT: Incorporating language structures into pre-training for deep language understanding. Proceedings of the 8th International Conference on Learning Representations. arXiv preprint arXiv: 1908.04577, 2020.

16 Qiu XP, Sun TX, Xu YG, *et al*. Pre-trained models for natural language processing: A survey. arXiv preprint arXiv: 2003.08271, 2020.

17 Qiao YF, Xiong CY, Liu ZH, *et al*. Understanding the behaviors of BERT in ranking. arXiv preprint arXiv: 1904.07531, 2019.

18 Nogueira R, Yang W, Cho K, *et al*. Multi-stage document ranking with BERT. arXiv preprint arXiv: 1910.14424, 2019.

19 Lu WH, Jiao J, Zhang RF, *et al*. TwinBERT: Distilling knowledge to twin-structured BERT models for efficient retrieval. arXiv preprint arXiv: 2002.06275, 2020.

20 Kim Y. Convolutional neural networks for sentence classification. arXiv preprint arXiv:1408.5882, 2014.

21 Joulin A, Grave E, Bojanowski P, *et al*. Bag of tricks for efficient text classification. Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics. Valencia, Spain. 2017. 427–431.

22 段丹丹, 唐加山, 温勇, 袁克海. 基于 BERT 模型的中文短文本分类算法.计算机工程, 2021, 47(01): 79–86.