多模型融合的客服工单文本分类方法的研究与实现

张亮[1]，代晓菊[1]，郑荣[1]，贺同泽[2]
（1. 上海理想信息产业（集团）有限公司，上海 201315；2. 北京邮电大学，北京 100876）

摘　要：电信呼叫中心客服在人工进行工单分类时存在归档耗时长、效率低、准确率难以保障的问题，但此场景下类别数量多，且类别间具有层级关联，导致传统文本分类方法准确率较低。针对此问题，提出了一种基于多模型融合的文本分类方法，根据不同层级的数据特点使用不同模型进行分类，考虑了类别的层级关联以提升准确率，并验证了此方法的有效性，可以优化客服生产系统运营流程，加快现场人工客服响应能效，提升客服热线整体运营效率，实现人工智能注智生产。

关键词：文本分类；客服工单；多模型融合；运营效率
中图分类号：TP391.1
文献标识码：A
doi: 10.11959/j.issn.1000−0801.2021236

Research and implementation of text classification method for customer service orders based on multi-model fusion

ZHANG Liang[1], DAI Xiaoju[1], ZHENG Rong[1], He Tongze[2]
1. Shanghai Ideal Information Industry (Group) Co., Ltd., Shanghai 201315, China
2. Beijing University of Posts and Telecommunications, Beijing 100876, China

Abstract: Due to the large amount of order categories and their hierarchical associations, traditional manual order classification method of customer service in telecom call center has the problems of long archiving time, low efficiency and unsustainable accuracy. To solve this problem, a novel text classification algorithm based on multi-model fusion was proposed, which intelligently classify orders with multiple models based on data characteristics and their hierarchical associations, the effectiveness of this method was verified. The current manual operation process was optimized and operation efficiency was enhanced, which support the intelligent transformation and upgradation of existing customer service system.

Key words: text classification, customer service order, multi-model fusion, operational efficiency

1　引言

目前国内传统的呼叫中心，主要通过人工客服和自助语音助手两种方式，全天候提供咨询、订购办理、投诉建议等热线服务。目前自助语音助手只能帮助用户进行简单的查询和订购操作，而面对复杂业务诉求，诸多用户还是会选择人工客服来处理诉求。人工客服不但要记住客户诉求，

收稿日期：2021−03−22；　修回日期：2021−10−14
基金项目：上海互联网大数据工程技术研究中心资助项目（No.15DZ2250700）
Foundation Item: Shanghai Internet Big Data Engineering Technology Research Center (No.15DZ2250700)

还要在相关业务运营系统进行分类归档，势必会耗费大量时间，完全采用人工的方式进行管理越来越不能满足实际的需求，并且伴随用户需求剧增和变化，对呼叫中心的接通率提出更高和更全面的要求，势必需要融入各种新技术来使之适应需求。

以电信客服呼叫中心为例，目前日均呼入量为 25 000 通，而接通率为 92%，电话诉求日均受理量已达到 20 000 多件，每位人工客服日均受理 80 通电话或录音。每通电话平均处理时长为 5.75 min，其中通话平均时长为 3.25 min，案头归档平均处理时长为 2.5 min。平台完全依赖人工、服务能力趋于饱和，人工客服工作强度大，服务效率难以提升，最终导致接通率难以保证。人工客服在受理诉求时，需要耗费大量时间去理解分析、手工整理用户诉求内容，并且工单业务分类体系复杂（多层级、多类别）、操作流程烦琐，十分依赖于人工客服的知识储备和经验能力，是造成电信客服呼叫中心运营效率产生瓶颈的主要原因。

为进一步提升呼叫中心服务渠道“对内智能辅助、对外智能服务”的能力，通过引入人工智能技术——中文文本自动分类，实现用户诉求的快速记录及工单准确分类归档。

本文针对电信客服呼叫中心人工客服在实际运营系统的重点、难点问题，结合并使用了目前若干前沿的中文本分类技术和算法，提出了一个多模型融合的客服工单文本分类方法，解决人工客服进行工单分类归档耗时长和准确率低的问题，为呼叫中心提供了一种自动快捷、高效准确的面向分类结构复杂的工单文本分类系统，优化客服生产系统运营流程，加快现场人工客服响应能效，提升客服热线整体运营效率。

目前随着机器学习、深度学习算法研究的逐渐深入，文本分类的方式、方法得到不断改进、优化，特别是在中文文本分类领域的研究成果日新月异，但很多研究的前沿技术都聚焦在较少类别的分类体系任务上，对超多类别且成层级体系的文本分类问题研究较少。多层级分类体系的特点在于不同层级的数据具有不同的规模，且层级间存在一定联系。然而，现有方法通常着重于挖掘层级间关联，对每个层级使用相同的模型，却忽略了每个层级的数据特点不同，单一模型并不适用于全部层级。因此，本文将基于电信客服领域工单文本，设计了多模型融合算法（multi model fusion algorithm，MMF），根据各层级的数据特点选择合适的模型，实现工单自动多级分类，以及提高分类准确度作为主要研究方向。

2　文本分类算法

中文文本自动分类是计算机对中文自然语言按照一定的分类体系或标准进行自动化归类、标记的过程，根据一个已标注的文档集合，训练学习得到文档特征和文档标签类别之间的关系模型，然后利用这种关系模型对新的文档进行类标签进行判断、预测。现针对本文实验过程中使用的算法如 XGBoost、TextCNN、HFT-CNN、BERT 做具体介绍。

2.1　XGBoost 算法

极度梯度提升（extreme gradient boosting，XGBoost）算法，是一种 boosted tree 的可扩展机器学习算法，经常被用在一些大型文本类比赛中，效果显著。它是目前最快最好的开源 boosted tree 工具包，能够高效、灵活和便利地进行大规模并行 boosted tree 算法操作。XGBoost 所应用的算法就是全梯度下降树（gradient boosting decision tree，GBDT）的改进，既可以用于分类也可以用于回归问题。XGBoost 算法的实现流程如下。

输入　训练样本 $I=\{(x1,y1),(x2,y2),\cdots,(xm,ym)\}$，最大迭代次数 T，损失函数 L，正则化系数 λ、γ；

输出　强学习器 $f(x)$；

迭代轮数：t=1,2,…,T；

步骤 1 计算第 i 个样本（i=1,2,…,m）在当前轮损失函数 L，计算所有样本的一阶导数和二阶导数。

步骤 2 基于当前节点尝试分裂决策树，默认分数值为 0，G 和 H 为当前需要分裂的节点的一阶二阶导数之和。

步骤 3 基于最大值对应的划分特征和特征值分裂子数。

步骤 4 如果最大值为 0，则当前决策树建立完毕，计算所有叶子区域的权重，得到弱学习器，更新强学习器，并进入下一轮弱学习器的迭代。如果最大值不为 0，则从步骤 2 继续尝试分裂决策树。

XGBoost 广泛应用最重要的原因是具有可扩展性。从工业界应用的落地情况来看，XGBoost 的分布式版本具有广泛的可移植性，支持在 MPI、YARN 等多个平台上运行，同时得益于 XGBoost 保留了单机并行版本的各种优化，使得工业界数据规模的问题可以得到很好的解决。

2.2 TextCNN 算法

TextCNN 算法模型结构是 CNN 结构的一个变体，如图 1 所示。

在对一系列句子进行分类时，假设 X_i 为句子 S 中的第 i 个词，$\boldsymbol{X}_i$ 是由 k 维词向量组成，表示为 $\boldsymbol{X}_i \in \mathbf{R}^k$，长度为 n 的句子（必要时加上空格）表示为：

$$\boldsymbol{X}_{1:n} = \boldsymbol{X}_1 \oplus \boldsymbol{X}_2 \oplus \cdots \oplus \boldsymbol{X}_n \tag{1}$$

其中，$\oplus$ 表示连接符，即输入矩阵的大小为 $n \times k$。一般来说，$\boldsymbol{X}_{i:i+j}$ 表示一系列词 $\boldsymbol{X}_i, \boldsymbol{X}_{i+1}, \cdots, \boldsymbol{X}_{i+j}$ 连接的 k 维词向量。

在进行卷积操作时，滤波器（作用于词窗产生新的特征）不再是横向滑动，仅仅是向下移动，类似于 N-gram 的思想来提取局部文本中词与词之间的相关性，“词窗”的含义是每次作用几个单词，反映在图上就是滤波器一次性遍历几行。图 2 所示中有 2、3、4 共 3 种步长策略，每个步长都有两个滤波器（实际训练时滤波器会很多）。滤波器可作用在不同的词窗上以产生卷积后的特征向量。例如，特征向量 $\boldsymbol{C}_i$ 是由词窗 $\boldsymbol{X}_{i:i+h-1}$ 生成的，计算式为：

$$\boldsymbol{C}_i = f(w \cdot \boldsymbol{X}_{i:i+h-1} + b) \tag{2}$$

其中，f 表示一个非线性函数，w 表示权重，b 表示偏移量。

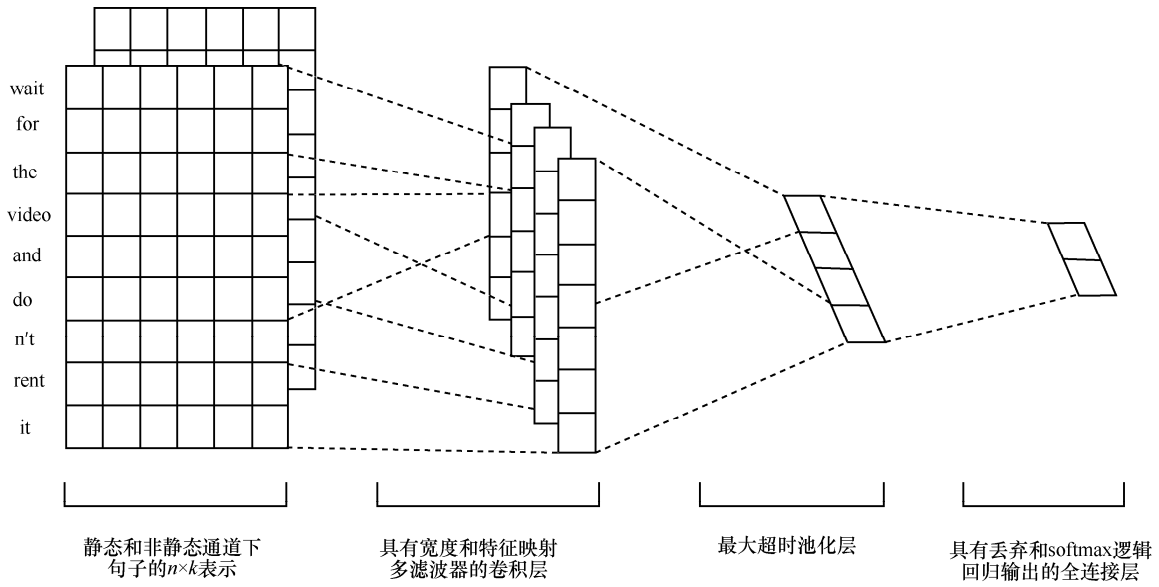

图 1 TextCNN 算法结构模型

然后对每一个特征向量进行最大化池化操作并拼接各个池化值，最终得到句子 S 的特征表示，将这个句子向量输入分类器进行分类，至此完成整个流程。

针对文本浅层特征的抽取 TextCNN 能力很强，在短文本搜索、对话等领域专注于意图分类时效果很好，并且速度快，应用较为广泛。由于 TextCNN 主要靠滤波器窗口抽取特征，针对长文本领域中的长距离建模方面能力受限，且对语序不敏感，因此应用不是特别突出。

2.3　HFT-CNN 算法

HFT-CNN 模型类似于 TextCNN，使用的是 fastText 进行词向量计算，输入是由词向量拼接而成的短文本句子序列，接着使用卷积核为 w 的卷积层提取句子特征，然后添加池化层，将这些池化层的结果拼接然后经过全连接层和 Dropout（降低过拟合的可能性进而提高模型的泛化能力）得到上层标签[A,B,···]的概率，损失函数采用交叉熵。

HFT-CNN 针对下层标签的预测是基于在上层标签的预测模型中已经学到了通用特征，但更深层的网络需要去学习原始数据集中比较详细的信息。因此模型中对词嵌入和卷积层参数保持不变，在这个基础上进行微调学习，这一步标签也由[A,B]变为[A1,A2,B1,B2]。采用了两种得分方式来对最终文本分类的结果进行判断：布尔评分函数（boolean scoring function，BSF）和乘法评分函数（multiplicative scoring function，MSF）。这两种得分方法都是设置一个阈值，预测文本在某个分类标签的得分超过阈值则认为是该分类标签。区别在于 BSF 只有在文本被分到一级分类标签时才会认为分类的二级分类标签是正确的，MSF 则没有此限制。对标签的分类，模型是先学习比较通用的特征知识，然后再进行细分。

HFT-CNN 模型标签层级化微调思路值得深入研究，但是也存在一些问题：一方面是关于差异性特征抽取问题；另一方面，按照遍历递进思路，多元标签数量多，层级也比较高，则模型会变得异常复杂，训练的速度很慢。这也是工业界未能广泛应用的重要原因。

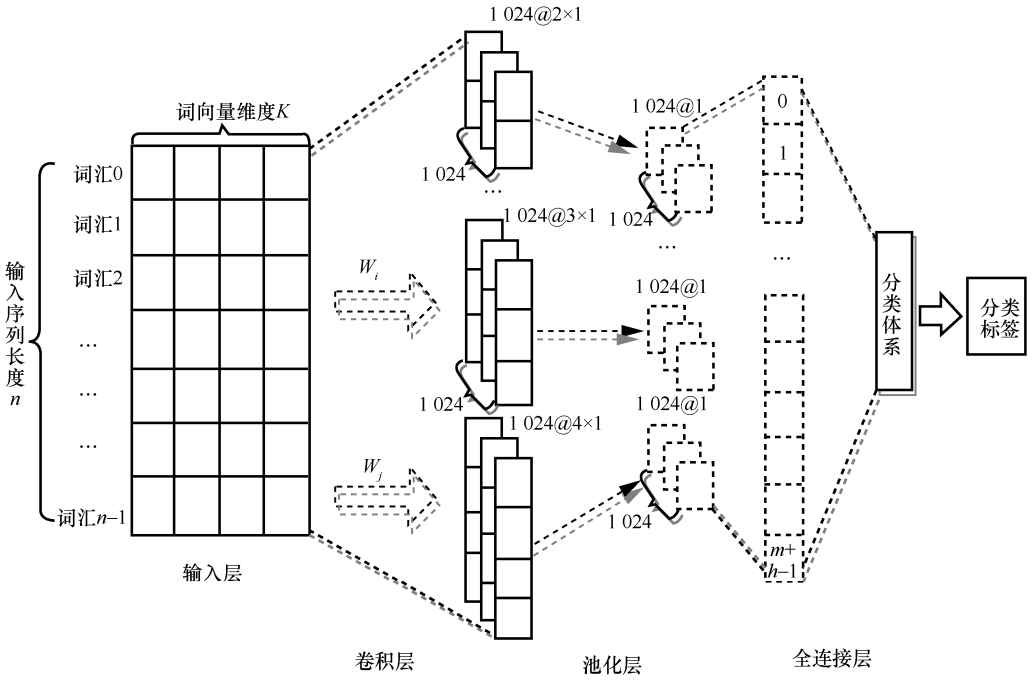

图 2　TextCNN 算法流程

2.4 BERT 算法

BERT（bidirectional encoder representation from transformers）算法模型是一种基于双向 Transformer 构建的语言模型。语言模型（language model）是指找到一串字或词序列的概率分布，可以表示一个句子或序列出现的概率，通过概率模型表示文本语义，从而可以量化地衡量一段文本存在的可能性，也就是这个句子或序列的逻辑是否通顺。对于一段长度为 N 的文本序列，序列里每个单字或词都有上文预测该单字或词的过程，所有单字或词的概率乘积可以用来评估文本存在的可能性。Transformer 是首个完全依靠自注意力机制来计算其输入和输出表示，而不使用序列对齐的循环或卷积的神经网络转换模型，旨在解决序列到序列的关联任务，同时能够轻松处理长时序依赖。此处“转换”是指将输入序列转换成输出序列。Transformer 创建的核心理念是通过注意力和重复机制，彻底处理输入和输出之间的依赖关系。BERT 的模型架构是一个多层双向 Transformer 编码器，且 BERT 模型中与 Transformer 相关的实现和原 Tranformer 结构几乎一样。BERT 的设计不同于单向语言模型（ELMo、GPT）来学习通用语言表征，它是受到完型填空任务的启发，通过在所有网络层中对单个字或词的左右上下文进行联合调节，将来自未标记文本的深层双向表示进行预先训练，减轻单向语言模型的约束问题。

使用 BERT 主要有两个步骤：预训练和微调，如图 3 所示。在预训练期间，BERT 模型的主要作用是在不同任务的未标记数据上进行训练学习；而微调的时候，BERT 模型是用预训练好的参数进行初始化，基于下游任务的已有标签的数据来训练学习的。尽管最初的时候都是用预训练好的 BERT 模型参数，但每个下游任务有自己的微调模型。模型将预训练模型和下游任务模型结合在一起，下游任务中仍然使用 BERT 模型，而且支持文本分类任务，在做文本分类任务时不需要对模型做修改。将 BERT 在大量的领域相关数据上继续训练，使得 BERT 更好地适应于相关领域的文本分布表示，最终在该领域的下游任务上取得进一步的提升。

很多实验证明 BERT 适合处理句子对匹配类的任务，在 GLUE（general language understanding evaluation）这种综合的 NLP 数据集下，对几乎所

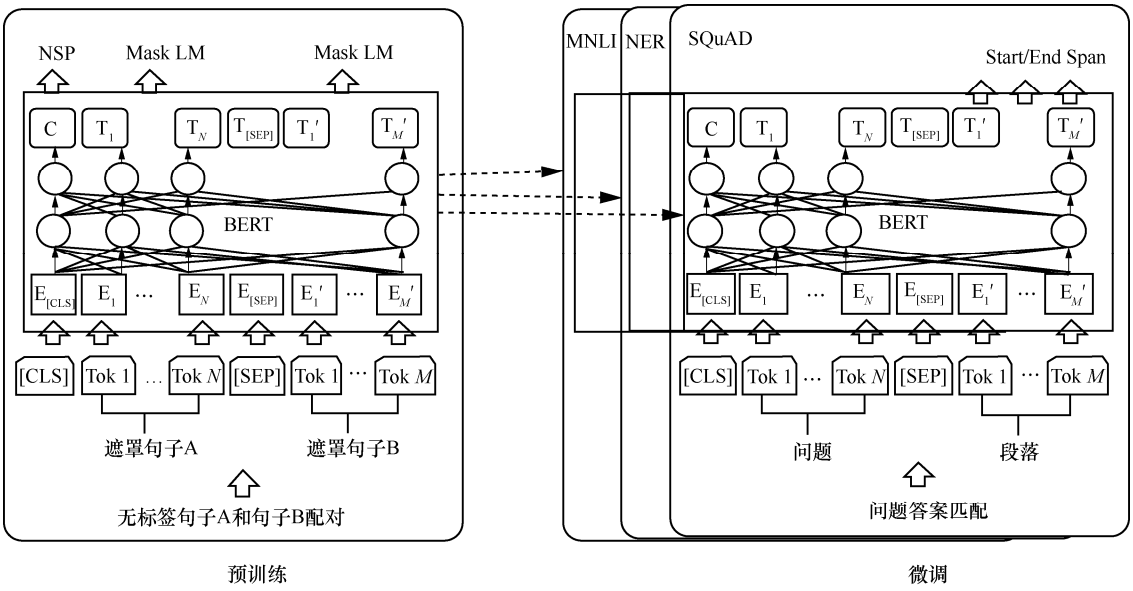

图 3 BERT 算法的预训练和微调

有类型的 NLP 任务（除了生成模型外），BERT 预训练都有明显促进作用。但是，GLUE 的各种任务有一定比例的数据集规模偏小，领域也还是相对有限，例如在书面文本领域表现较好，而对于口语文本领域的表现就相对较差。随着 BERT 本身能力的各种增强，绝大多数 NLP 子领域都会被统一到 BERT 两阶段+Transformer 特征抽取器的方案框架上，预训练技术对于很多应用领域的确产生了很大的促进作用。

3　多模型融合算法分类方法

3.1　基本思路

由于单级的文本分类系统在分类类别特别多时，受到类别空间大、数据稀疏性的影响，准确率较低，因此现有的针对超多类别的文本分类系统多为基于分级结构的方法。然而，现有的基于分级结构的方法多为每级使用相同的机器学习模型或深度学习模型，但忽略了不同的层级上的数据特点不同，适合的模型也不同，导致部分层级准确率较低。为了验证这一观点，本文使用 TextCNN 在四级分级结构的每一层直接进行分类，结果见表 1。第一级和第二级的类别个数相对较少，且每一类别的数据量也较为充足，TextCNN 在这两级的也有着较高的准确率。而第三级的类别个数则多至上百个，每个类别的数据量有明显的降低，在第四级这种现象进一步加剧，导致 TextCNN 模型的准确率很低。因此本文提出了基于分级结构的多模型融合算法，结构如图 4 所示，提升分类准确率。

表 1　各级分类数据及直接使用 TextCNN 分类准确率情况

层级	类别数	平均每个类别数据量	TextCNN 准确率
第一级	9	8 942	0.9
第二级	68	1 183	0.819
第三级	256	314	0.674
第四级	654	123	0.601

为了充分利用标签的分级特性，解决深度模型在靠后层级上分类准确率下降的问题，将分类模块设计为两个子模块，分别为粗分类模块和细分类模块，分类流程如图 5 所示。粗分类模块的

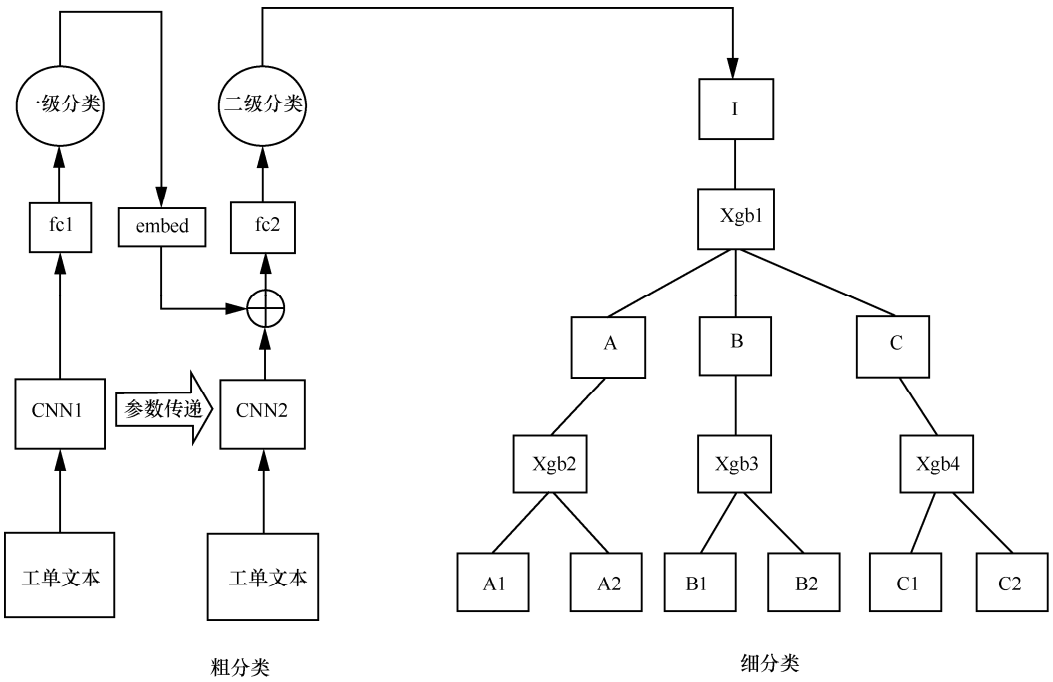

图 4　MMF 模型整体结构

目的是将文本分类到非最后一层的某一层叶节点类别上，由于数据量充足且类别空间较小，可以采用深度学习模型进行分类，本文采用 TextCNN 作为分类器。细分类模块则是将粗分类得到的结果继续沿着分级结构向下分类，直至分类到最后一层类别节点上，由于分类空间变大，数据更加稀疏，所以本文采用针对每一个类别训练分类器的方式，并使用对数据量要求较低的 XGBoost 作为分类器。

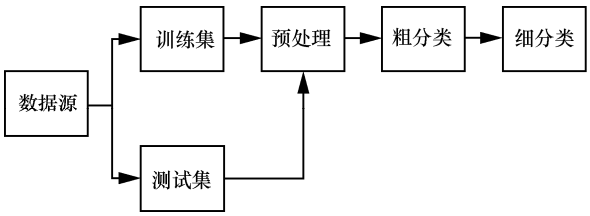

图 5　分类流程

3.2　基于 TextCNN 的粗分类

粗分类模块流程如图 6 所示，以粗分类级数为 2 级为例。在粗分类模块中，将文本分类到较高层的类别，由于分类空间相对最后一集较小，所以很多深度学习方法都可以使用，选择了经典的 TextCNN 模型。

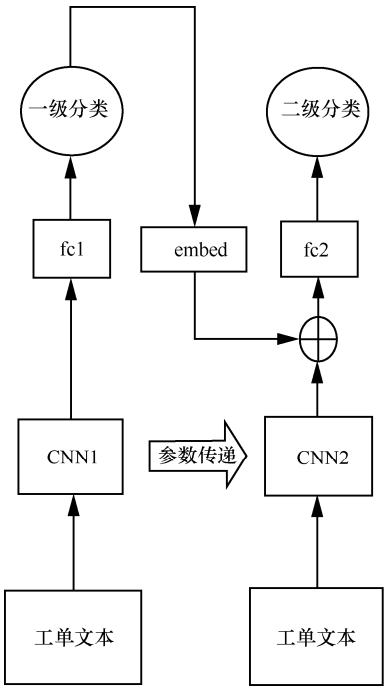

图 6　粗分类模块流程

然而，接近百种类别的类别空间仍然会导致模型受到数据稀疏性的影响，对于 TextCNN 这种含有大量的参数的模型而言，在小数据集上训练会极大地影响泛化能力，通常会导致过度拟合，为缓解这种影响，本文采用微调的技术训练模型。微调的动机是观察到 TextCNN 上层可以捕捉到对许多任务都有效的通用的特性，TextCNN 后面的层则逐渐变得更与任务相关联，以捕捉到原始数据集中包含的细节。这个动机与将类别标签做分级的动机是相同的，因为本文首先将不同的类别在层次结构的上层进行相对粗粒度的区分，然后在底层进行更加细粒度的区分，也就是说，上层的标签更加具有泛化性。因此在层级之间对 TextCNN 做微调，可以充分利用类别标签的层级结构。具体的方法是，将第一层 TextCNN 模型的上层参数，包括词向量、卷积层、池化层传递到第二层的 TextCNN 模型，然后对下层参数，即全连接层使用训练数据微调，以此类推。

考虑到前面几级的类别标签数量较少，且具有较强正交性，即类别之间区分较为明显，对后续的分类有更强的影响。如果第一级模型将文本分类为交通类，那第二级的模型应该更关注在交通类的子类别中进行细分，又因为模型在前面几级分类准确率较高，所以本文采取在后续层级的分类中引入上一级模型向量的方式来辅助当前层级模型的学习。图 6 中，将第一级 TextCNN 模型的最后一层向量与第二级 TextCNN 模型的最后一层向量做拼接后，经过全连接层输出分类类别。如果粗分类包括多于两级的话，则对其他级也做如上操作。

3.3　基于 XGBoost 的细分类

经过粗分类后，每个样本被分类到了中间层的某个类别上。这时，如果继续用粗分类的框架继续分类，则会面临分类空间巨大，数据非常稀疏的情况，深度学习方法由于需要足够数据量支撑，不再适用，即使通过微调的方式缓解，模型

的准确度依然不高。为解决这个问题，本文采用针对每个类别单独训练模型的方式，而不是使用一个模型分类所有类别。这样做的优势在于每个模型的分类空间变小，分类难度降低。但同时，考虑到每个模型对应的训练数据也变少了，使用对训练数据量要求较小的 XGBoost 算法。相比于深度学习算法，XGBoost 有较强的非线性拟合能力，在训练数据较少时仍有比较好的记忆性，且只占用 CPU 资源，可以同时进行多个模型的训练和预测。

XGBoost 算法需要人为构造特征作为输入。通过观察数据发现，多数类别是具有较为明显的关键词的，所以首先将文本数据集分词并且除去停用词以后，进行文本特征词的提取。在文本的特征词提取中，关注的是某个词与类别是否存在比较强的相关性。如果是，那么这个词就具有表征此类文章的能力，则可以作为此类文本的特征词。为了表示某个词与某个类别的相关性，本文使用 CHI 检验方法提取特征词。通过给每一类文本选出 150 维的特征并去重，可以获得 1 000 维左右的特征，接着为每个文本构造向量空间模型，即使用词频-逆文件频率（TF-IDF）的方法计算各特征的权重得到表示该文本的特征向量，就将原本的单个文本转化成了 1 000 维特征向量，对应一个分类类别号这种可以方便使用 XGBoost 算法分类的数据。

在某一层进行分类后，每条样本都会得到对应分类类别的概率分布，常用的是类别选择方式是贪心搜索，即将概率最高的类别当作此条样本在这一层的类别，下一层在这个类别的基础上继续分类，但由于在下一级是针对每个类别的子类别进行细分，层级之间具有强依赖关系，这种贪心搜索的方式只能保证局部最优而不能保证全局最优，如果在上一级分错，那下一级的分类结果也一定是错的。为了缓解这个问题，本文采用束搜索的方式，通过挑选概率最高的两个类别，然后在下一层用这两个类别对应的子模型进行分类，每个子模型同样保留概率最高的两个类别，当分类到最后一层时，每个分类类别都对应着一条路径，将这个路径上对应的概率相乘，将乘积最大对应的类别作为最终的分类结果。如图 7 所示，加粗的路径为按照上述逻辑选出来的路径。

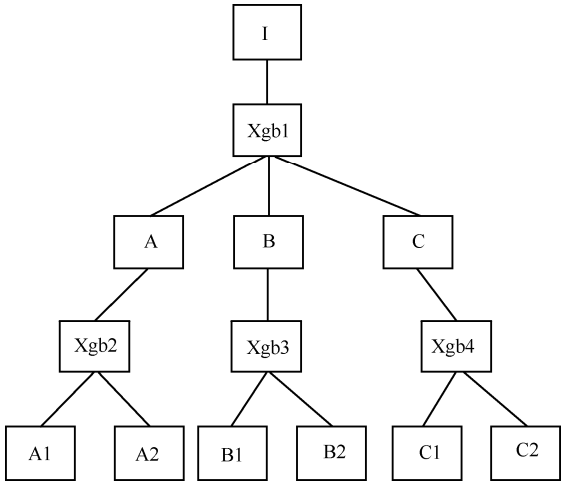

图 7　细分类模块流程

4　实验结果及分析

4.1　实验数据

为了验证基于分级结构的多模型融合工单分类方法的可行性和优势，本文将其与现有的文本分类模型进行验证和比对。实现了基于分级结构的多模型融合工单分类方法包括文本预处理模块和分类模块，分类模块又可以分为基于 TextCNN 的粗分类和基于 XGBoost 的细分类两个子模块。文本样本输入系统后，首先进行文本预处理，处理为适合下游模型输入的数据格式，常用手段包括过滤非中文信息、模板提取和文本分词等。然后输入分类模块开始进行分类，得到分类结果。

实验验证数据集为客服中心某套餐业务 2018 年、2019 年的工单数据。数据集包含 80 482 条样本，将其划分为训练集和测试集，训练样本 72 000 条，测试样本 8 482 条，数据集的标签体系分为四级，每级的类别数和数据情况见表 1。

数据预处理基于 Python3/SKLearn 实现；模型训练和验证，基于 Python3/TensorFlow1.14 实现。

4.2 实验环境

实验环境硬件采用的是 P40 显卡（运行内存 24 GB），软件开发环境为：CUDA（10.1）、cuDNN（7.4）、Anaconda（4.10.1）、Python（3.6）、Tensoflow-gpu（1.14）、Scikit-learn（0.23.2）。

4.3 评估方法

实验分别验证了 TextCNN、分级 TextCNN、BERT、分级 BERT、XGBoost、分级 XGBoost、HFT-CNN 和本文提出的 MMF 共 8 种模型在数据集上的准确率（所有预测四级分类全部正确的样本/总样本）。对于 TextCNN 模型，设置其卷积核窗口大小为（2，3，4），步长为 1；对于 BERT 模型，使用与论文中相同的超参数设置。对于 XGBoost 模型，由于数据量较少容易导致过拟合，对其进行了较为细致的调参，设置树模型最大深度为 5，树的个数为 100，特征子采样比率为 0.7，L1 正则化参数为 0.5，L2 正则化参数为 1，学习率为 0.05。

4.4 实验结果和分析

4.4.1 实验结果对比

模型实验对比结果见表 2，通过对比可以看出，直接用单一模型分类所有类别的 TextCNN、BERT 和 XGBoost 方法准确率较低，BERT 相比于 TextCNN 的提升也十分有限。而基于分级的方法，分级 TextCNN、分级 BERT、分级 XGBoost、HFT-CNN 和本文提出的方法则明显优于非分级方法，说明在类别空间非常大的时候，利用标签的分级体系进行分类是非常有必要的。其中，分级 BERT 和 HFT-CNN 比分级 XGBoost 方法的准确率更高，说明进行级别之间的参数共享是可以进一步缓解数据稀疏性的。

本文提出的 MMF 明显超过了其他对比方法，相比分级 BERT 方法高出 3.8%，相比 HFT-CNN 方法高出 3.9%，相比分级 XGBoost 方法高出 5.1%，相比分级 CNN 高出 6.3%。主要原因在于使用了粗分类和细分类的分类架构，前者利用数据量较为充足的优势，使用分级的 TextCNN 模型进行分类，并且为了充分利用层级结构，本文使用了层级之间 TextCNN 模型的参数共享。后者采用对每个子类别单独分类的方法，用训练更多模型换取更高的精度，同时使用对数据量要求较小的 XGBoost 方法作为分类器，减轻过拟合的风险。在粗分类的时候，同时还将第一级模型的最后一层的向量加入到后续层级的模型中，使得后续层级分类时更关注在其父类类别上。此外，细分类时本文采用束搜索的方式来获得全局最优的分类效果。

表 2 模型试验对比结果

模型	预处理	第一级	第二级	第三级	第四级	最终
TextCNN	分词+词向量	—	—	—	—	0.601
分级 TextCNN	分词+词向量	0.901	0.910	0.902	0.841	0.622
BERT	词向量	—	—	—	—	0.619
分级 BERT	词向量	0.893	0.903	0.884	0.907	0.647
XGBoost	特征提取	—	—	—	—	0.582
分级 XGBoost	特征提取	0.867	0.880	0.912	0.916	0.634
HFT-CNN	分词+词向量	0.898	0.908	0.886	0.894	0.646
MMF	分词+词向量+特征提取	0.901	0.910	0.912	0.916	0.685

本文在粗分类里将第一级模型的最后一层的向量加入后续层级的模型中，提高粗分类的分类精度，在细分类中采用束搜索的方式来获得全局最优的分类效果，为了验证这两个操作的有效性，本文分别针对这两个操作进行消融实验，实验结果如图 8 所示，模型嵌入（model embedding，ME）和束搜索（beam search，BS）分别代表着模型分别将这两个操作去掉的实验结果，可以看到，粗分类中将第一级模型向量加入后续模型可以带来 1.48%的收益，这说明第一层模型的分类效果对后续模型有着较大的影响，通过让后续模型关注在父类类别的子类别上，可以有效缩小分类空间，提升粗分类精度。此外，细分类中的束搜索可以带来 0.93%的收益，这说明与贪婪搜索相比，束搜索通过扩大搜索空间，可有效减轻单个模型精度不足导致的错分情况。

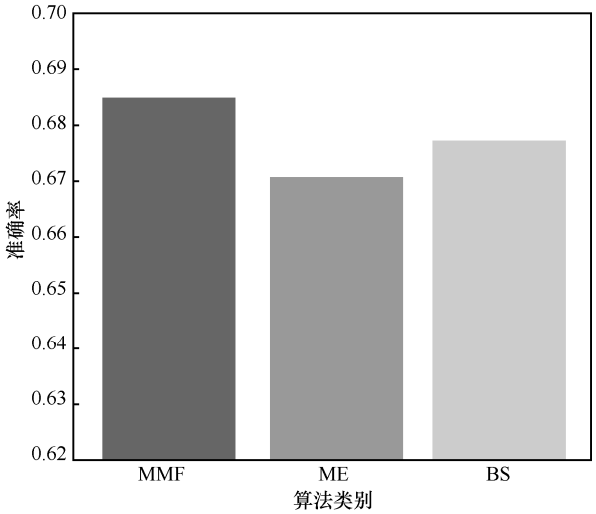

图 8　模型向量和束搜索的有效性验证

4.4.2　分级方法性能比较

为了更好地体现本文提出的分类框架的优越性，本文将每一级的准确率指标与同为分级方法的分级 TextCNN、分级 BERT、分级 XGBoost、HFT-CNN 进行比较，其结果如图 9 所示。与分级 BERT、分级 XGBoost 方法相比，本文提出的方法在前两级的分类效果上有明显提升，原因在于模型的粗分类模块是用的分级 TextCNN，准确率要好于分级 XGBoost 算法；在后两级上，本模型的表现也好于另外 3 种分级的方法（分级 TextCNN、分级 BERT、分级 XGBoost），这说明了针对每个子类单独训练 XGBoost 模型可以有效缓解分类空间大的问题，并且使用 XGBoost 作为分类器有效防止了数据量少带来的过拟合问题。

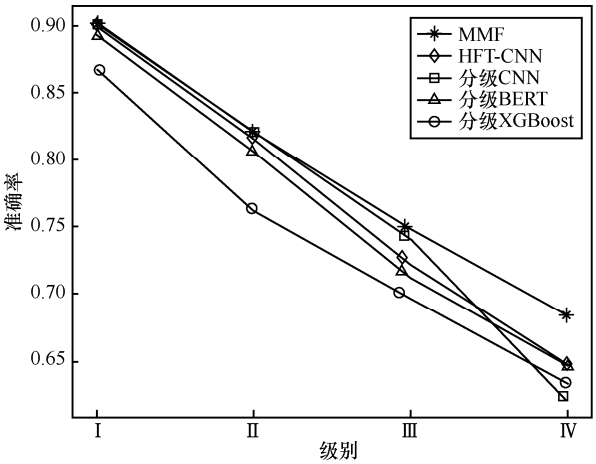

图 9　分级方法性能比较

5　结束语

文本分类一直有着广泛的应用需求，然而现有的方法在针对分类类别数目较少的情况时性能较好，当分类空间增大时，算法的准确性会降低。

由于文本的多层级的特殊性，即使每层的分类器准确率超过 90%，受到多层级分类器累积的影响，最终准确率值并未达到一个较高的水平。后续建议在提高语音转文本准确率的同时，加强数据预处理操作（包括停用词、同义词、纠错等方式）提高输入数据的质量，同时希望能结合领域专家知识，重新梳理工单本身的分类结构体系，保证工单类别尽可能精确、独立。

本文在分析了现有方法劣势的基础上，提出了一个基于分级结构的多模型融合的工单分类方法，包括粗分类和细分类两个子分类模块，前者使用层级 TextCNN 获得粗粒度的分类，后者通过对每个子类别学习多个树分类器，进行

更精细的分类。通过在实际生产数据集上的实验验证可发现，本文提出的分类框架可有效提升分类效果。

参考文献：

[1] KIM Y. Convolutional neural networks for sentence classification[C]//Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP). USA: Association for Computational Linguistics, 2014.

[2] MA M B, HUANG L, ZHOU B W, et al. Dependency-based convolutional neural networks for sentence embedding[C]//Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 2: Short Papers). USA: Association for Computational Linguistics, 2015.

[3] Devlin J, Chang M W, Lee K, et al. BERT: pre-training of deep bidirectional transformers for language understanding[EB]. 2018.

[4] ZHAO Z W, WU Y Z. Attention-based convolutional neural networks for sentence classification[C]//Interspeech 2016. [S.l:s.n.], 2016.

[5] SHIMURA K, LI J Y, FUKUMOTO F. HFT-CNN: learning hierarchical category structure for multi-label short text categorization[C]//Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing. USA: Association for Computational Linguistics, 2018.

[6] CHEN T Q, GUESTRIN C. XGBoost: a scalable tree boosting system[C]//Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. New York: ACM Press, 2016: 785-794.

[7] ZHANG Z Y, HAN X, LIU Z Y, et al. ERNIE: enhanced language representation with informative entities[C]//Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics. USA: Association for Computational Linguistics, 2019.

[8] KALCHBRENNER N, GREFENSTETTE E, BLUNSOM P. A convolutional neural network for modelling sentences[C]// Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers). USA: Association for Computational Linguistics, 2014.

[9] YIN W P, SCHÜTZE H. Multichannel variable-size convolution for sentence classification[C]//Proceedings of the Nineteenth Conference on Computational Natural Language Learning. USA: Association for Computational Linguistics, 2015.

[作者简介]

张亮（1991− ），男，上海理想信息产业（集团）有限公司软件产品研发工程师，主要研究方向为人工智能技术、自然语言处理、大数据挖掘与分析。

代晓菊（1990− ），女，上海理想信息产业（集团）有限公司软件产品研发工程师，主要研究方向为人工智能技术、自然语言处理、大数据挖掘与分析。

郑荣（1981− ），男，上海理想信息产业（集团）有限公司软件产品研发高级工程师，主要研究方向为人工智能技术、自然语言处理、大数据挖掘与分析。

贺同泽（1996− ），男，北京邮电大学硕士生，主要研究方向为推荐系统、自然语言处理、大数据挖掘与分析。