基于预训练语言模型的案件要素识别方法

刘海顺[1] **王雷**[2] **陈彦光**[1] **张书晨**[1] **孙媛媛**[1†] **林鸿飞**[1]
[1]大连理工大学计算机科学与技术学院，辽宁大连 116024
[2]辽宁省人民检察院第三检察部，辽宁沈阳 110033
dahai@mail.dlut.edu.cn, 18804002266@163.com
{cygariel, camael}@mail.dlut.edu.cn, syuan@dlut.edu.cn, hflin@dlut.edu.cn

摘要

案件要素识别指将案件描述中重要事实描述自动抽取出来，并根据领域专家设计的要素体系进行分类，是智慧司法领域的重要研究内容。基于传统神经网络的文本编码难以提取深层次特征，基于阈值的多标签分类难以捕获标签间依赖关系，因此本文提出了基于预训练语言模型的多标签文本分类模型。该模型采用以Layer-attentive策略进行特征融合的语言模型作为编码器，使用基于LSTM的序列生成模型作为解码器。在“CAIL2019”数据集上进行实验，该方法比基于循环神经网络的算法在F1值上最高可提升7.6%，在相同超参数设置下比基础语言模型（BERT）提升约3.2%。

关键词： 案件要素识别；多标签文本分类；智慧司法；语言模型

A Method for Case Factor Recognition Based on Pre-trained Language Models

Haishun Liu[1] **Lei Wang**[2] **Yanguang Chen**[1] **Shuchen Zhang**[1]
Yuanyuan Sun[1†] **Hongfei Lin**[1]
[1]School of Computer Science and Technology，Dalian University of Technology
[2]The Third Procuratorial Department，People’s Procuratorate of Liaoning Province
dahai@mail.dlut.edu.cn, 18804002266@163.com
{cygariel, camael}@mail.dlut.edu.cn, syuan@dlut.edu.cn, hflin@dlut.edu.cn

Abstract

Case factor recognition is an important research content in the domain of legal intelligence. The purpose of this task is to automatically extract the important fact descriptions from the legal case descriptions and classify them based on the factor system designed by the domain experts. Text encoding based on traditional neural networks is difficult to extract deep-level features, and threshold based multi-label classification is difficult to capture the dependencies between labels. So that a multi-label text classification model based on pre-trained language models is proposed. The encoder is the language model fine-tuned with the strategy of Layer-attentive, and the decoder is LSTM based sequence generation model. Experimented on the CAIL2019 dataset, the method can improve the F1 score by up to 7.6% over the traditional neural network algorithm based on Recurrent Neural Network, and about 3.2% over the basic language model under the same hyperparameter settings.

Keywords: Case factor recognition , Multi-label text classification , Legal intelligence , Language model

[†]通信作者

1 引言

2018年，司法部印发《“十三五”全国司法行政信息化发展规划》，明确提出我国到2020年全面建成智能高效的司法行政信息化体系3.0版，将大数据、人工智能、云计算、物联网等技术与司法工作进行实际融合，实现公共法律服务的便捷普惠化，实现政务管理水平的高效透明。随着我国司法行政信息化的不断推进，智慧司法研究领域兴起并日趋火热。智慧司法包括法律阅读理解、案件要素识别、相似案例匹配和司法判决预测等任务，旨在赋予机器理解法律文本的能力，促进司法智能的发展。其中，案件要素识别的具体研究内容为，给定裁判文书中的相关段落，针对文书中每个句子进行判断，识别其中的关键案情要素。案件要素抽取的结果不仅可以为要素式裁判提供技术支持，还可以应用到案情摘要、可解释性的类案推送以及相关知识推荐等司法领域的实际业务需求中。

前人在司法智能领域的研究工作主要集中在司法判决预测 (Luo et al., 2017; Zhong et al., 2018; Hu et al., 2018)、相似案例匹配和命名实体识别测 (Wang, 2018; Xie, 2018)等方面，直接针对案件要素识别的研究还相对较少，但它们在技术上具有共通性。与通用领域的自然语言处理（NLP）任务 (Kim, 2014; Bahdanau et al., 2014; Yang et al., 2016)类似，当前研究者们在智慧司法领域采用的方法多是基于神经网络的结构。具体地，网络底层使用预训练的词向量进行词嵌入，中层采用卷积神经网络（Convolutional Neural Networks, CNN）或者循环神经网络（Recurrent Neural Network, RNN）提取特征，上层应用分类器进行分类或应用条件随机场（Conditional Random Field，CRF）进行序列标注。这种结构存在一定的缺点，一是使用的静态词向量无法处理不同语境下的一词多义问题(Peters et al., 2018)，二是有监督方法的本质致使模型性能受限于标注数据集的大小。

不同于一般的多分类问题，案件要素识别是多标签分类问题，即一个样本可能同时属于0到N个类别。经统计分析，不计算负例，每个样本平均包含2.7个标签，最多可达7个。而且多个类别之间往往具有关联性，如Figure 1所示，在离婚类案件中，若一个样本属于“限制行为能力抚养子女”类，那么该样本有较大概率同时属于“婚后有子女”类，在借贷类案件中，“有借贷证明”多和“有书面还款协议”一起出现。解决多标签分类问题的主流方法是将其处理为多个二分类问题(Boutell et al., 2004)，通过设定阈值判断样本是否属于每个类。这种方法明显忽略了标签之间的相关性，性能有限。

针对上述问题，本文专门就案件要素识别任务进行了研究，提出了基于预训练语言模型的案件要素多标签分类方法。预训练语言模型支持上下文有关的词嵌入，可以从庞大的无标注数据中学习丰富的语法、语义等特征表示，捕获更长距离的依赖。BERT(Devlin et al., 2019)是预训练语言模型的一个基础模型，于公布之初横扫了11项NLP任务。结合Yang (2018)的工作，本文将BERT系列语言模型作为案件要素识别整体模型的编码器，且提出了Layer-attentive的多层特征的融合策略，将长短期记忆网络（Long Short-Term Memory，LSTM）作为解码器，并对比了与基于阈值算法的多标签分类的性能差异。最后，在公开的CAIL2019“要素识别”数据集上验证了模型的性能。

裁判文书句子描述	案件要素类别
原告王某某诉称：x 年 x 月 x 日，原、被告在 x 市民政局协议离婚，离婚时约定，长子王某甲由被告抚养，长女王某乙由原告抚养。	婚后有子女； 限制行为能力子女抚养
原告规划院向本院提出诉讼请求：1、判令原告不支付被告解除劳动合同的赔偿金 58288 元；	经济性裁员
原告诉称，被告侯 x 在 x 年 x 月 x 日向原告出具书面个人借款承诺书，提出向 x 银行贷款 70000 元，请原告担保。	借款金额 x 万元； 有借贷证明； 贷款人系金融机构； 有书面还款承诺

Figure 1: 案件要素识别实例

第十九届中国计算语言学大会论文集，第744页-第754页，海南，中国，2020年10月30日至11月1日。

2 相关工作

智慧司法研究由来已久。早在上世纪五、六十年代，研究者就开始通过数学统计的方法对司法案件进行定量分析(Kort, 1957; Ulmer, 1963)，随后在八、九十年代，研究者们探索了基于规则的专家系统(Shapira, 1990; Hassett, 1993)。随着机器学习技术的发展，司法判决预测作为智慧司法研究的主要任务而备受关注，基于支持向量机（Support Vector Machine, SVM）的预测模型被提出来，预测对象包括罪名、案件类别和裁判日期等(Aletras et al., 2016; Sulea et al., 2017)。近年来，由于司法数据的公开和深度学习的发展，我国在司法判决预测方面出现了许多瞩目的工作。Luo (2017)通过BiGUR（双向门控神经单元）建模判决书文档及法条信息以进行罪名预测，CAIL2018(Xiao et al., 2018)提出了第一个用于司法判决预测的大规模中文法律数据集，Zhong (2018)以CNN和LSTM为基础构建了同时预测罪名、法条和刑期的多任务学习模型，Hu (2018)通过引入司法属性研究了少数罪名的预测问题。案件要素识别是司法智能领域的新兴任务，现阶段主要被当做文本分类问题进行处理，在技术上与司法判决预测最接近。

作为案件要素识别核心技术的文本分类，近几年，主流方法逐步从词向量加神经网络向语言模型转变。2013年开始，Word2Vec(Mikolov et al., 2013)以网络结构简单、易于理解、使用方便等特征成为最流行的词向量训练工具之一。随后，Kim (2014)结合词向量提出了多维度并行的单层卷积神经网络，模型表现优于传统机器学习方法和早期神经网络方法。紧接着，RNN也被引入文本领域，其变体LSTM(Hochreiter and Schmidhuber , 1997)以能捕获长距离信息依赖、善于编码序列信息而得到大量应用，Yang (2018)提出了基于LSTM序列生成模型的多标签文本分类算法。而后注意力机制被广泛研究(Bahdanau et al., 2014; Yang et al., 2016)，Lin (2017)提出Self-attentive，通过二维矩阵对序列信息进行加权。2018年，谷歌的研究人员提出了基于自注意力机制对Transformer框架(Vaswani et al., 2017)，并以Transformer为核心组件开发出了性能强大的语言模型BERT。

BERT的预训练及微调方法被不断进行改进 (Yang et al., 2019; Cui et al., 2019; Liu et al., 2019)。Qiao (2019)提出的BERT(MUL-Int)将每一层的[CLS]位置的编码进行加权求和，进而计算索引问题和答案文档之间的相似度。Sun (2019)基于BERT设计了更多的实验，不仅验证了每一层输出对分类结果的影响，还提出以简单平均的方式融合前四层或后四层输出。本文基于以上提到的文本分类模型进行了案件要素识别的相关实验和分析，对比了不同语言模型的性能差异，在Lin (2017)、Qiao (2019)和Sun (2019)等人工作的基础上提出了Layer-attentive特征融合策略。就多标签文本分类而言，本文使用LSTM序列生成模型，并对比了与阈值算法的性能差异。

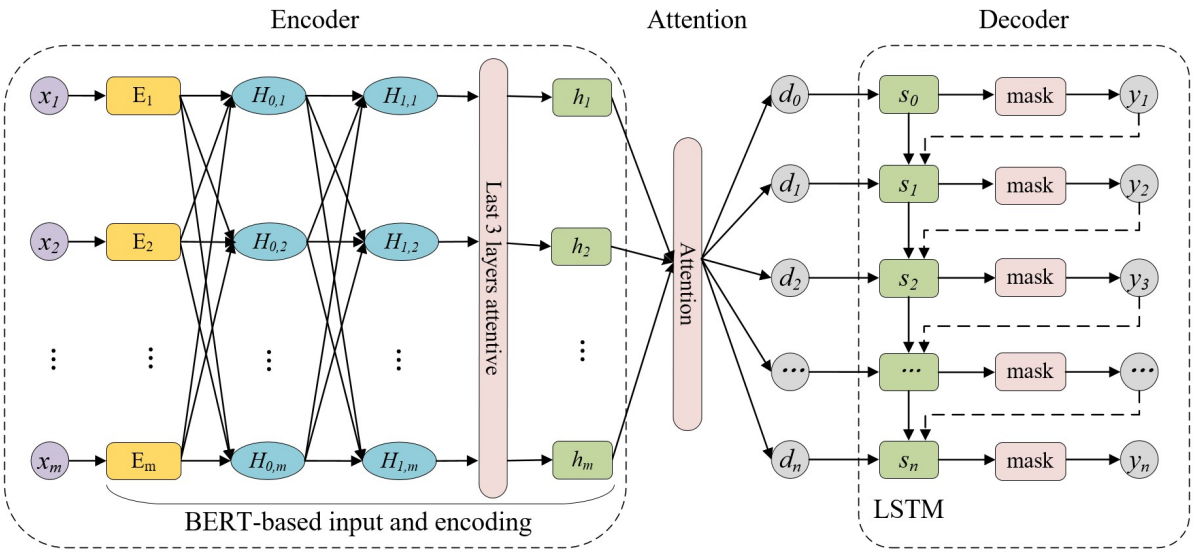

Figure 2: 基于预训练语言模型的案件要素识别模型

3 方法

本文将案件要素识别任务形式化描述为一个多标签分类任务，给定裁判文书中的一个句子序列$X = \{x_1, x_2, ..., x_m\}$，预测与$X$相对应的案件要素类别集合$\tilde{Y} \subseteq Y$。其中$m$是序列$X$的长度，$x_i$表示序列中的第$i$个词。$Y = \{y_1, y_2, ..., y_n\}$为要素标签集合，$n$为要素类别总数，因为一个样本可能同时归属于多个要素类别，所以$\tilde{Y}$是Y的子集。

本文所构建模型的结构如Figure 2所示，整体主要包括编码器（Encoder）和解码器（Decoder），中间通过注意力机制（Attention）操作进行交互。编码器部分以BERT为主体，将BERT最后三层的输出以Layer-attentive的方式进行加权融合得到输出H，对$H = [h_1, h_2, ..., h_m]$和$S = [s_1, s_2, ..., s_m]$进行Attention操作得到$D = [d_0, d_1, ..., d_n]$，将D通过基于LSTM的解码器得到标签预测集合$\tilde{Y}$。

3.1 BERT预训练语言模型

下面以BERT(Devlin et al., 2019)为例介绍BERT系列的预训练语言模型。BERT预训练语言模型的全称是基于Transformer的双向编码表示（Bidirectional Encoder Representations from Transformers, BERT）。其采用Transformer网络(Vaswani et al., 2017)作为模型基本结构，在大规模无监督语料上通过掩蔽语言模型和句对预测两个任务进行预训练（Pre-training），得到预训练的BERT模型。再以预训练模型为基础，在下游相关NLP任务上进行模型微调（Fine-tuning）。BERT模型的结构主要由三部分构成：输入层、编码层和任务层，其中输入层和编码层是通用的结构，对任何任务都适用。

BERT的输入层将每个词的词嵌入、位置嵌入和段嵌入相加得到每个词的输入表示$[E_1, E_2, ..., E_m]$。与原始Transformer不同的是，BERT模型的位置嵌入是可学习的参数，最大支持长度为512个位置。

对于编码层，base版本包含12层编码层，large版本包含24层编码层，每一层的输入都是基于上一层的输出，可抽象表示为：

$$H_i = \text{Transformer}(H_{i-1}), 0 < i < l \tag{1}$$

其中，$H_i \in R^{m \times d}$，m为序列长度，d为隐层维度。

在本任务中，任务层被Attention交互层和解码器替代。

3.2 基于预训练模型的编码器

一个神经网络的不同层可以捕获不同的语法和语义信息。因为BERT包含了l（12或24）个编码层，研究表明(Qiao et al., 2019; Sun et al., 2019)，选择BERT后3至4个编码层的输出进行特征融合，可以增强语言模型的特征表示。本文提出了Layer-attentive，以层次级别加权的方式对后三个编码层的输出进行融合，公式如下：

$$A_i = \text{softmax}(W_2 \text{tanh}(W_1 H_i^T)) \tag{2}$$

$$H = \text{SeLu}(A_{-1}H_{-1} + A_{-2}H_{-2} + A_{-3}H_{-3}) \tag{3}$$

其中，$W_1 \in R^{d \times d}$，$W_2 \in R^{d \times d}$，是两个权重矩阵，可以将向量的表示聚焦于不同层的不同元素。SeLU(Klambauer et al., 2017)是非线性激活函数。本文将以上特征融合方法命名为3Lattv。

为了证明以上方法的有效性，本文还设计了其他的特征融合方法。一是采用concat的方式对后三层的输出进行线性拼接：

$$H' = \text{SeLU}(W_c(H_{-1} \oplus H_{-2} \oplus H_{-3}) + b_c) \tag{4}$$

其中$W_c \in R^{d \times 3d}$，$\oplus$表示线性拼接。该方法被命名为3Lconcat。二是在上述两种方法中改后三层为后四层，相应的方法被命名为4Lattv和4Lconcat。

3.3 注意力交互

当模型预测不同的标签时，并非所有文本词都作出相同的贡献。Attention通过关注文本序列的不同部分并聚集那些信息丰富的词的隐层表示来产生上下文向量。特别地，注意力在时间步t上将权重α_{ti}分配给第i个单词，如下所示：

$$e_{ti} = v_a^T \text{tanh}(W_a s_t + U_a h_i + b_a) \tag{5}$$

$$\alpha_{ti} = \frac{\exp(e_{ti})}{\sum_{j=1}^{m} \exp(e_{tj})} \tag{6}$$

其中，W_a，U_a，v_a是权重参数，b_a是偏置项，s_t是解码器在时间步t的当前隐藏状态。在时间步t传递到解码器的最终上下文向量d_t的计算如下：

$$d_t = \sum_{i=1}^{m} \alpha_{ti} h_i \tag{7}$$

其中d_t的物理意义是预测第t个标签时的解码器的输入。

3.4 基于LSTM的解码器

本文使用LSTM(Hochreiter and Schmidhuber , 1997)作为多标签分类的解码器，解码器在时间步t的隐藏状态s_t的计算公式如下：

$$s_t = \text{LSTM}(s_{t-1}, y_{t-1}, c_{t-1}) \tag{8}$$

其中，y_{t-1}是时间步*t-1*在标签空间Y上的概率分布，其计算如下：

$$o_t = W_o \sigma(W_d s_t + U_d c_t + b_d) \tag{9}$$

$$y_t = \text{softmax}(o_t + I_t) \tag{10}$$

其中W_o，W_d，U_d是权重系数，$I_t \in R^Y$是用于防止解码器预测重复标签的掩码向量，即Figure 2中mask部分，σ是非线性激活函数。如果标签y_i在第*t-1*时间步被预测出来，则$I_t = -\infty$，否则$I_t = 0$。

最后，使用交叉熵损失函数进行训练：

$$J(\theta) = -\frac{1}{Nn} \sum_{i=1}^{N} \sum_{j=1}^{n} (y_{ij} \log p_{ij} + (1 - y_{ij}) \log(1 - p_{ij})) \tag{11}$$

其中N为样本个数，n为标签个数，y为实际标签，p为预测标签。

4 实验

4.1 数据集介绍

本文实验使用CAIL2019“要素识别”赛道提供的数据集[0]，该数据集来自“中国裁判文书网”公开的法律文书，由专家进行标注。数据的每一条由一个句子及其对应的要素标签组成，句子是从一篇裁判文书中的部分段落提取出来的，实例如Figure 1所示。本文将其按3:1:1的比例划分训练集、开发集和测试集，在测试集上评价模型性能。数据集涉及三类民事案件：劳动争议（Labor）、离婚纠纷（Divorce）和借贷纠纷（Loan），三类案件的数据各自分开，分别进行评价。每类案件各有20个要素类别，相应的类别样本数分布如Figure 3所示。可见数据集存在严重的数据分布不均衡的问题，每个案件的要素类别样本数从10_1级到10_3级不等。数据集的样本数据量统计及在样本的文本特点分析见Tanle 1。另发现平均60%以上的样本没有标签，即不是案件要素；一个样本最多可有7个标签，此种情况不足0.1%；具有1到3个标签的样本在三类案件中分别占约30%、25%、37%。

Table 1: 数据集的样本数量统计表

案件	案例数	每案例平均样本数	样本平均长度	样本数			
				训练集	开发集	测试集	合计
Labor	836	37.95	57.22	19038	6346	6346	31730
Divorce	1269	29.09	48.07	22152	7384	7384	36920
Loan	634	35.73	74.43	13615	4538	4538	22691

4.2 环境及参数设置

本文所有实验在如Table 2所示的环境下进行。对于BERT系列模型，均采用base-

[0]https://github.com/china-ai-law-challenge/CAIL2019

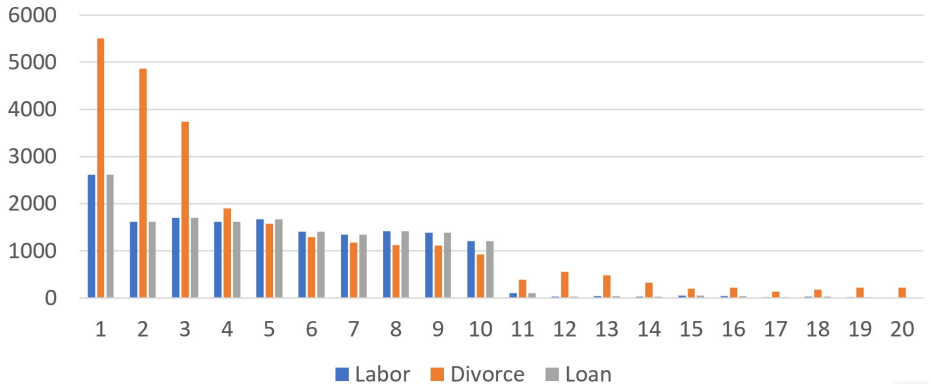

Figure 3: 各要素类别的数据量分布统计图

Table 2: 实验环境

环境名称	配置
操作系统	Ubuntu 16.04
CPU	Intel(R) Xeon(R) CPU E5-2678 v3 @ 2.50GHz
GPU	NVIDIA Tesla K80@11GB
Python	3.7.5
Pytroch	1.3.1
内存	128G

Chinese版本[1]进行微调，隐层维度d=768，序列长度m=512，编码层层数l=12，批处理大小batch_size=16，训练轮数epoch=3，学习率α=4e-5。对于BiLSTM模型，设置隐层维度hidden_size=256，序列长度m=256，学习率α=1e-3，批处理大小batch_size=64，训练轮数epoch=128，采用Word2Vec预训练的词向量的维度为300。

4.3 结果及分析

对于模型的表现，使用查准率(Precision, P)、查全率(Recall, R)和F1值作为衡量指标。具体使用宏平均查准率(Macro Precision, ma-P)、宏平均查全率(Macro Recall, ma-R)、宏平均F1值(Macro F1, ma-F)、微平均F1值(Micro F1, mi-F)、ma-F和mi-F的均值(Average F1, Ava)[0]。

4.3.1 编码器的作用

分别采用不同的编码器模型和解码器LSTM进行组合，在三个案件的数据上均进行实验。编码器模型列表如下：

BERT[1]：基础模型(Devlin et al., 2019)。

CNN-thre：Kim (2014)提出的卷积神经网络模型，底层使用预训练的词向量，使用多重一维卷积和最大池化提取特征。不使用解码器，输出层采用Algorithm 1所述方法。

BiLSTM：双向LSTM(Hochreiter and Schmidhuber , 1997)网络，底层使用预训练的词向量。

WWM[2]：基于Whole Word Masking训练样本生成策略训练的BERT(Cui et al., 2019)。

XLNet[3]：基于Transformer-XL(Dai et al., 2019)训练的最优自回归语言模型(Yang et al., 2019)。

RoBERTa[2]：采用多种技巧及更多数据训练的BERT(Liu et al., 2019)。

Table 3展示了在使用解码器LSTM的情况下，不同编码器模型在三类案件数据上的实验结果。比较CNN-thre、BiLSTM和BERT三个模型，BiLSTM优于CNN-thre，BERT优

[1]https://github.com/huggingface/transformers
[2]https://github.com/ymcui/Chinese-BERT-wwm
[3]https://github.com/ymcui/Chinese-XLNet

于BiLSTM，但该优势对语言模型而言提升并不特别明显。原因是一方面任务数据量达到万级，BiLSTM也能充分学习文本特征，另一方面训练BiLSTM所依据的词向量是根据数百万份裁判文书预训练的，Word2Vec在这里起到了很大的作用。为了详细比较BiLSTM和BERT在每个类别上的分类能力，Figure 4给出了BERT 和BiLSTM在Loan案件数据上每个要素类别的F1值。Figure 4表明，BERT对每个类别的分类能力均高于BiLSTM，在后10个类别，BERT的性能提升比较明显，结合Figure 3可知，Loan数据的后10个类别的样本数较前10个类别有数量级级别的差距，该结果也表明，以BERT为代表的语言模型处理小样本情况的能力较强。

纵向比较后四个模型，即四个BERT系列语言模型。BERT作为基础模型，性能较更先进的语言模型有一定的差距，XLNet和RoBERTa在该任务上具有最好的性能。RoBERTa比CNN-thre这一baseline模型绝对提升7.6%。另外，ma-F得分远低于mi-F得分，原因是数据分布极不均衡，每个类别的F1值相差很大，甚至有样本数量极少的类别的得分是0，这对ma-F影响较大，却对mi-F影响不明显。

Table 3: 不同编码器模型在三类案件数据上的实验结果

模型	Labor			Divorce			Loan		
	mi-F	ma-F	Ava	mi-F	ma-F	Ava	mi-F	ma-F	Ava
CNN-thre	75.03	50.26	62.65	80.05	66.34	73.20	72.08	47.03	59.56
BiLSTM	77.67	52.58	65.13	81.82	68.83	75.33	75.36	49.79	62.58
BERT	80.35	56.06	68.21	84.70	72.06	78.38	78.70	53.81	66.26
WWM	80.54	56.58	68.56	82.80	72.22	77.51	78.60	53.68	66.14
XLNet	81.24	58.48	69.86	**85.04**	**76.61**	**80.81**	79.33	55.77	67.55
RoBERTa	**81.29**	**58.66**	**69.96**	84.21	74.39	79.30	**79.82**	**57.06**	**68.44**

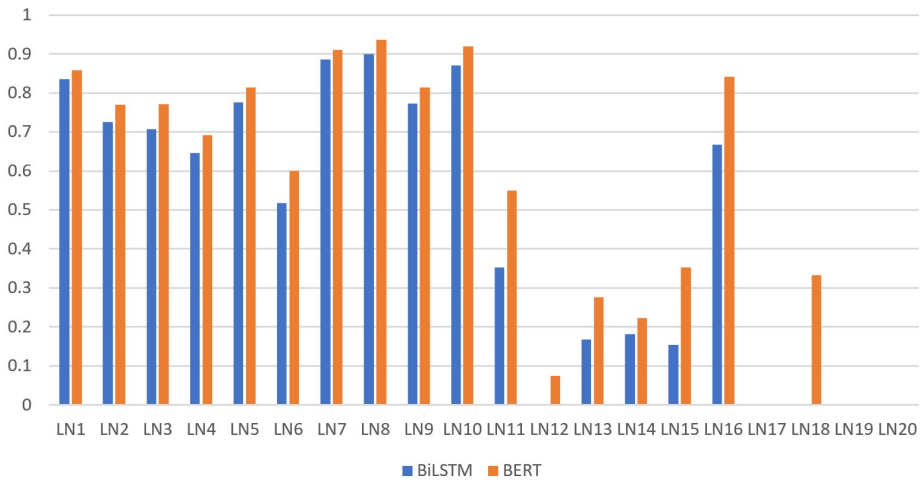

Figure 4: Loan数据上两个模型的类别F1值对比

4.3.2 解码器的作用

BERT-thre：该方法为只使用基于BERT的编码器，不使用Attention交互和解码器，相应的任务层换为softmax分类器，最后使用阈值设定函数对模型输出的概率值进行取舍，从而预测类别，标签概率计算公式如下：

$$p = \sigma(W_p \text{Pooler}(\text{H}) + b_p) \tag{12}$$

其中$W_p \in R^{d \times d}$，σ为sigmoid激活函数，Pooler是BERT对隐层输出进行pooling操作的函数(Devlin et al., 2019)。p中每个维度的数值对应每个类别的可能概率值，概率值介于[0,1]之间，仍使用二进制交叉熵损失函数进行训练。

Algorithm 1 类别阈值选择算法(Threshold selecting, thre)

Input: 在开发集上，样本属于第i类标签的概率p_i，样本在第i类的真实标签y_i。
Output: 第i类标签的阈值t_i。

t=arr[100], f=arr[100], s=arr[90], t[0]=0;
for j in 100 **do**
    t[j] = t[j-1] + 0.01;
    if $p >$ t[j] **then**
        判断当前阈值t[j]下样本类别$\hat{a}_i = 1$;
    else
        $\hat{a}_i = 0$;
    计算当前阈值t[j]下的类别得分f[j] = F($\hat{y}_i, y_i$);
    j递增1;
for k in 90 **do**
    保存每个区间下的得分均值s[k] = ($\sum_{l=0}^{9}$ f[$k+l$])/10;
    k递增1;
找到使得分最大的阈值区间z = argmax(s[z])，计算该区间的中值t_i = t[$z+5$];
return t_i

本文Algorithm 1所示算法为每个类别设定阈值。

由于多标签分类的特殊性，具体的P、R值只能通过两者的宏平均或者微平均来体现，Tabel 4通过比较模型在三类案件数据上的ma-P、ma-R和ma-F，具体验证解码器对P值和R值带来的提升。

Table 4: 解码器与阈值算法的实验结果对比

模型	Labor			Divorce			Loan		
	ma-P	ma-R	ma-F	ma-P	ma-R	ma-F	ma-P	ma-R	ma-F
BERT-thre	57.41	53.10	55.47	71.87	72.63	71.95	54.15	51.52	53.14
BERT-LSTM	57.96	53.79	56.06	71.74	72.90	72.06	54.62	52.38	53.81
提升	+0.55	+0.69	+0.59	-0.13	+0.27	+0.11	+0.47	+0.86	+0.67
RoBERTa-thre	60.22	56.19	58.43	75.83	74.62	74.46	57.46	55.30	56.67
RoBERTa-LSTM	69.43	56.46	58.66	75.82	74.50	74.39	57.81	55.74	57.06
提升	+0.21	+0.27	+0.23	-0.01	-0.12	-0.07	+0.35	+0.44	+0.39

在Table 4中，同一案件下编码器较thre策略的主要提升体现在R值（召回率）上，尤其对Loan案件最为明显，经分析如Loan中两个要素类别“贷款人系金融机构款”和“有借贷证明”之间的相关性达到了0.729，其他类别也具有明显的相关性。基于LSTM的解码器正因为捕获了这种相关性，才在预测出来一个标签的情况下能连带着把与之相关的标签也预测出来。但是，准确率增益差说明这种解码器也存在不足，标签预测过程中会出现一定的错误累积，前一个标签预测错误可能导致后一个相关的标签预测错误，后续研究工作中将着重在这方面进行改进。RoBERTa-LSTM相对BERT-thre的提升为3.2%。

4.3.3 Layer-attentive策略的作用

为验证多层特征融合策略对模型性能的影响，以及对比不同的融合方法，本组对比实验以原始BERT为基础模型，在此基础上分别使用3Lattv、3Lconcat、4Lattv和4Lconcat的方法进行实验，五种方法均采用基于LSTM的解码器，不同方法在三类数据上的得分见Table 5。

由Table 5可知，除BERT-4Lconcat方法外，其他多层特征融合方法优于原始BERT的方法。其次，除Labor案件下三层特征融合外，Layer-attentive的方法均优于concat线性拼接的方法，最大提升可达到2.1%。分别比较BERT(4Lconcat)和BERT(3Lconcat)，比

Table 5: Layer-attentive策略的作用

模型	Labor			Divorce			Loan		
	mi-F	ma-F	Ava	mi-F	ma-F	Ava	mi-F	ma-F	Ava
BERT	79.45	54.79	67.12	84.13	72.64	78.39	78.23	45.21	61.72
BERT(4Lconcat)	81.22	53.40	67.31	84.44	71.75	78.10	76.42	49.78	63.10
BERT(4Lattv)	79.08	55.75	67.41	84.29	72.24	78.27	79.03	49.43	64.23
BERT(3Lconcat)	80.50	56.08	**68.29**	84.37	71.87	78.12	76.50	51.82	64.16
BERT(3Lattv)	80.35	56.06	68.21	84.70	72.06	**78.38**	78.70	53.81	**66.26**

较BERT(4Lattv)和BERT(3Lattv)，可发现三层特征融合均优于四层特征融合。最后，对三类案件的得分进行横向比较，相同模型在三类案件上性能差异明显，主要原因是三类案件的数据量有一定差距，而且分别具有不同的要素类别体系。

4.3.4 模型案例分析

Figure 5所示为BiLSTM、BERT、WWM、WWM-LSTM四种模型分别对三类案件预测结果的例子。第一个例子为Labor案件，实际标签有三个，BiLSTM模型预测出0个，BERT只能预测出其中一个，而WWM可以预测出来两个标签，WWM-LSTM因为能捕获LB3和LB6之间的依赖关系，可以将三个标签全部预测出来。说明语言模型相对传统神经网络具有更强的学习能力。WWM因为考虑了中文分词问题，比原始的BERT具有更强的语义解析能力。第二个例子也展示了WWM更强的性能。第三个例子是Loan案件，原本句子没有标签，BiLSTM却错误地预测了一个标签，因为句子中含有“债权”关键字，BiLSTM只捕获了这个特征，没有理解语义信息，而语言模型的强大之处在于不仅能捕获浅层的语法特征，更能学习到深层的语义信息。

裁判文书句子描述	标签	BiLSTM	BERT	WWM	WWM-LSTM	标签含义
因被告不给原告签订劳动合同也不交纳社保，为此原告要求解除合同并支付经济赔偿金。	['LB1','LB3', LB6']	[]	['LB1']	['LB1', LB6']	['LB1', 'LB3', LB6']	LB1：解除劳动关系 LB3：支付经济补偿金 LB6：未签订劳动合同
故对原告诉请赵二×由原告抚养的请求本院予以支持。	['DV1', 'DV2']	[]	[]	['DV2']	['DV1', 'DV2']	DV1：婚后有子女 DV2：限制行为能力子女抚养
另查明，二审中，信×公司自认其因参与主债务人江×公司破产债权分配，已分配得到债权金额为×元。	[]	['LN1']	[]	[]	[]	LN1：债权人转让债权

Figure 5: 不同模型的预测结果示例

5 结束语

本文提出了一个基于预训练语言模型的多标签分类模型，该模型可实现面向司法领域的案件要素识别。该模型主要分为编码器和解码器两大部分，两部分间通过注意力机制进行交互，其中编码器部分采用基于Layer-attentive特征增强的语言模型，解码器采用LSTM序列生成模型。实验结果表明，本文提出的案件要素识别模型比基于循环神经网络的模型在F1值上提高了7.6%，比基础语言模型BERT提升约3.2%。本文采用的基于LSTM的多标签分类策略具有较大的性能增益，Layer-attentive的微调策略也有一定的性能提升。未来工作将研究要素类别的含义对要素识别结果的影响。

致谢

本文工作受国家重点研发计划（2018YFC0830603）资助。

参考文献

Bingfeng Luo, Yansong Feng, Jianbo Xu, Xiang Zhang, and Dongyang Zhao. 2017. Learning to predict charges for criminal cases with legal basis. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, 2727-2736.

Haoxi Zhong, Zhipeng Guo, Cunchao Tu, Chaojun Xiao, Zhiyuan Liu, and Maosong Sun. 2018. Legal Judgment Prediction via Topological Learning. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, 3540-3549.

Zikun Hu, Xiang Li, Cunchao Tu, Zhiyuan Liu, and Maosong Sun. 2018. Few-shot charge prediction with discriminative legal attributes. In *Proceedings of the 27th International Conference on Computational Linguistics*, 487-498.

Limin Wang. 2018. *Research on Chinese Named Entity Recognition for Legal Documents.* Suzhou University.

Yun Xie. 2018. *Reserch on Naming Entry Recognition for Chinese Legal Texts.* Nanjing Normal University.

Yoon Kim. 2014. Convolutional Neural networks for sentence classification. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing*, 1746-1751.

Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. 2014. Neural machine translation by jointly learning to align and translate. *arXiv preprint arXiv* :1409.0473.

Zichao Yang, Diyi Yang, Chris Dyer, Xiaodong He, Alex Smola, and Eduard Hovy. 2016. Hierarchical attention networks for document classification. In *Proceedings of the 2016 conference of the North American chapter of the association for computational linguistics: human language technologies*, 1480-1489.

Matthew E. Peters, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, and Luke Zettlemoyer. 2018. Deep contextualized word representations. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, 2227-2237.

Jacob Devlin, Ming W. Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of Deep Bidirectional Transformers for Language Understandin. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, 2227-2237.

Pengcheng Yang, Xu Sun, Wei Li, Shuming Ma, Wei Wu, and Houfeng Wang. 2018. SGM: Sequence generation model for multi-label classification. In *Proceedings of the 27th International Conference on Computational Linguistics*, 3915-3926.

Fred Kort. 1957. Predicting Supreme Court decisions mathematically: A quantitative analysis of the "right to counsel" cases. *American Political Science Review 51*, no. 1: 1-12.

Sidney S. Ulmer. 1963. Quantitative Analysis of Judicial Processes: Some Practical and Theoretical Applications. *Law and Contemporary Problems*, 28(1):164-84.

Monica Shapira. 1990. Computerized decision technology in social service. *International Journal of Sociology and Social Policy*, 10, 138–164.

Patricia Hassett. 1993. Can Expert System Technology Contribute to Improved Bail Decisions? *International Journal of Law and Information Technology*, 1(2):144-144.

Nikolaos Aletras, Dimitrios Tsarapatsanis, Daniel Preoţiuc-Pietro, and Vasileios Lampos. 2016. Predicting judicial decisions of the European Court of Human Rights: A natural language processing perspective. *PeerJ Computer Science*, 2:e93.

Octavia M. Sulea, Marcos Zampieri, Shervin Malmasi, Mihaela Vela, Liviu P. Dinu, and Josef V. Genabith. 2017. Exploring the use of text classification in the legal domain. *arXiv preprint arXiv* :1710.09306.

Chaojun Xiao, Haoxi Zhong, Zhipeng Guo, Cunchao Tu, Zhiyuan Liu, Maosong Sun, and Yansong Feng. 2018. Cail2018: A large-scale legal dataset for judgment prediction. *arXiv preprint arXiv* :1807.02478.

Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013. Efficient estimation of word representations in vector space. *arXiv preprint arXiv* :1301.3781.

Sepp Hochreiter and Jürgen Schmidhuber. 1997. Long short-term memory. *Neural computation*, 9(8): 1735-1780.

Zhouhan Lin, Minwei Feng, Cicero Nogueira dos Santos, Mo Yu, Bing Xiang, Bowen Zhou, and Yoshua Bengio. 2017. A structured self-attentive sentence embedding. *arXiv preprint arXiv* :1703.03130.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in neural information processing systems*, 5998-6008.

Zhilin Yang, Zihang Dai, Yiming Yang, Jaime Carbonell, Russ R. Salakhutdinov, and Quoc V. Le. 2019. Xlnet: Generalized autoregressive pretraining for language understanding. In *Advances in neural information processing systems*, 5754-5764.

Yiming Cui, Wanxiang Che, Ting Liu, Bing Qin, Ziqing Yang, Shijin Wang, and Guoping Hu. 2019. Pre-training with whole word masking for chinese bert. *arXiv preprint arXiv* :1906.08101.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. Roberta: A robustly optimized bert pretraining approach. *arXiv preprint arXiv* :1907.116921.

Yifan Qiao, Chenyan Xiong, Zhenghao Liu, and Zhiyuan Liu. 2019. Understanding the Behaviors of BERT in Ranking. *arXiv preprint arXiv* :1904.07531.

Chi Sun, Xipeng Qiu, Yige Xu, and Xuanjing Huang. 2019. How to fine-tune BERT for text classification? *China National Conference on Chinese Computational Linguistics*, 194-206. Springer, Cham, 2019.

Günter Klambauer, Thomas Unterthiner, Andreas Mayr, and Sepp Hochreiter. 2017. Self-normalizing neural networks. *Advances in neural information processing systems*, 971-980.

Matthew R. Boutell, Jiebo Luo, Xipeng Shen, and Christopher M. Brown. 2004. Learning multi-label scene classification. *Pattern recognition 37*, no. 9: 1757-1771.

Zihang Dai, Zhilin Yang, Yiming Yang, Jaime Carbonell, Quoc V. Le, and Ruslan Salakhutdinov. 2019. Transformer-xl: Attentive language models beyond a fixed-length context. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, 2978-2988.