·开发研究与工程应用·　　文章编号：1000-3428(2022)06-0288-07　　文献标志码：A　　中图分类号：TP391

面向法律文书的分段式摘要模型

王　刚，孙媛媛，陈彦光，林鸿飞

（大连理工大学 计算机科学与技术学院，辽宁 大连 116024）

摘　要：文本摘要是指对文本信息内容进行概括、提取主要内容进而形成摘要的过程。现有的文本摘要模型通常将内容选择和摘要生成独立分析，虽然能够有效提高句子压缩和融合的性能，但是在抽取过程中会丢失部分文本信息，导致准确率降低。基于预训练模型和Transformer结构的文档级句子编码器，提出一种结合内容抽取与摘要生成的分段式摘要模型。采用BERT模型对大量语料进行自监督学习，获得包含丰富语义信息的词表示。基于Transformer结构，通过全连接网络分类器将每个句子分成3类标签，抽取每句摘要对应的原文句子集合。利用指针生成器网络对原文句子集合进行压缩，将多个句子集合生成单句摘要，缩短输出序列和输入序列的长度。实验结果表明，相比直接生成摘要全文，该模型在生成句子上ROUGE-1、ROUGE-2和ROUGE-*L*的F1平均值提高了1.69个百分点，能够有效提高生成句子的准确率。

关键词：司法摘要；预训练模型；Transformer编码器；序列标注；指针生成器网络；分段式摘要模型

开放科学（资源服务）标志码（OSID）：

中文引用格式：王刚，孙媛媛，陈彦光，等. 面向法律文书的分段式摘要模型[J]. 计算机工程，2022，48(6)：288-294.

英文引用格式：WANG G，SUN Y Y，CHEN Y G，et al. Segmented summarization model for legal documents[J]. Computer Engineering，2022，48(6)：288-294.

Segmented Summarization Model for Legal Documents

WANG Gang，SUN Yuanyuan，CHEN Yanguang，LIN Hongfei

（School of Computer Science and Technology，Dalian University of Technology，Dalian，Liaoning 116024，China）

【Abstract】Text summary refers to the process of summarization the content of text information，extracting the relevant content，and then formulating a summarization.Existing text summarization models usually analyze content selection and summarization generation separately. Although they can effectively improve the performance of sentence compression and fusion models，some text information will be lost in the extraction process，resulting in reduced accuracy.Based on the pre-training model and the document level sentence encoder with Transformer structure，a segmented summarization model combining content extraction and summarization generation is proposed.The BERT model is used to conduct self-supervised learning on a large corpus to arrive at a word representation containing rich semantic information.Based on the Transformer structure，each sentence is divided into three types of tags through the fully connected network classifier，and the original sentence set corresponding to each sentence summarization extracted.The Pointer-Generator（PG）network is used to compress the original sentence set，and multiple sentence sets generated into a single sentence summarization to shorten the length of the output and input sequences.The experimental results show that compared with the direct generation of summarization full text，the F1 average value of ROUGE-1，ROUGE-2，and ROUGE-*L* in generating sentences increased by 1.69 percentage points，which can effectively improve the accuracy of generating sentences.

【Key words】judical summarization；pre-training model；Transformer encoder；sequence labeling；Pointer-Generator（PG）network；segmented summarization model

DOI：10. 19678/j. issn. 1000-3428. 0061119

0　概述

司法摘要是对司法文书的内容进行压缩、总结的过程，在案件检索、案件预览等场合中将司法摘要与业务流程相结合，能够提高司法工作的效率。在摘要任务中，与新闻、文学文本相比，法律文书的格

基金项目：国家重点研发计划（2018YFC0830604）。

作者简介：王　刚（1994—），男，硕士研究生，主研方向为自然语言处理；孙媛媛（通信作者），教授、博士生导师；陈彦光，硕士研究生；林鸿飞，教授、博士生导师。

收稿日期：2021-03-15　　**修回日期：**2021-07-15　　**E-mail：**syuan@dlut.edu.cn

式和结构更规范，用词更精确，在摘要的研究中具有明显的领域特点。司法领域数据建设的不断进步为司法领域的信息自动化、信息化、智能化建设提供了坚实的数据基础，同时也为自然语言处理（Natural Language Processing，NLP）方向的研究提供了新思路。

文本摘要是指机器自动对文本信息进行选择、凝练，进而形成较短文本。文本摘要分为抽取式摘要和生成式摘要。抽取式摘要是将抽取原文本中包含重要信息的句子组成摘要，句子间不够连贯，灵活性差。生成式摘要是对词库中词语的组合排列形成概括性的新句子，比较灵活，符合自然摘要的生成方式。生成式模型存在序列重复、可读性差等问题，随着深度神经网络的发展，其各个问题逐渐被解决，从而受到广泛关注。生成式摘要的可读性较优，但需要训练大量语料，大部分模型在处理长文本时存在训练时间过长和“长距离依赖”等问题。抽取式摘要和生成式摘要都可以对原文信息进行甄选，后者能够将信息重新组合成上下衔接更加流畅的句子。近年来，研究人员侧重于对内容选择和摘要生成进行独立分析。文献[1]提出选择合适的单句或句对生成单句摘要的方法。针对长文档摘要，文献[2]在生成摘要模型前增加简单的抽取步骤。研究人员通过对重要段落或句子的总结进行内容选择。相比摘要生成，内容选择更关注信息的识别，并且具有领域特点，例如，体育新闻注重比分、参赛运动员信息等，法律文本注重案情要素、法条等。摘要生成则侧重于语义上的裁剪和抽象。将内容选择和摘要生成分开能够有效测试句子压缩和融合模型的性能，并具有一定的可解释性，但是在抽取过程中可能会丢失一部分信息。

本文提出一种基于序列标注的分段式抽取摘要模型。在摘要抽取过程中，采用预训练模型作为句子嵌入和Transformer结构[3]的序列标注模型，将每个句子分成3类标签，抽取每句摘要对应的原文句子集合。在摘要生成过程中，利用指针生成器（Pointer-Generator，PG）网络[4]对抽取结果进行压缩和融合，生成单句摘要。

1　相关工作

自动摘要研究以抽取式方法为主，使用统计学方法，例如，以句子位置、词频、线索词等特征作为输入，并将得到每个句子的分数从大到小进行排序，抽取出分数较高的句子作为摘要[5-7]。随着机器学习技术在NLP中的应用，将统计特征输入到机器学习模型中进行分类的方法成为研究热点。文献[8]提出将文章的统计特征输入到朴素贝叶斯分类器模型和决策树的方法。文献[9]提出将马尔科夫模型用到摘要抽取中。文献[10-11]提出图排序的方法等。以上方法都是基于人工提取的浅层特征，然而基于深度神经网络的方法通过模型自学习识别文本的高维特征，采用序列标注的方法，即为每个句子定义0或1的标签，以包含1标签的句子作为被选中的重要句子。文献[12]基于循环神经网络（Recurrent Neural Network，RNN）的序列分类模型构建SummaRuNNer模型，将抽取摘要的内容重要度、显著度、新颖度等显式地表达出，具有较优的可解释性。文献[13]结合打分和选择提出NeuSum模型，是一种基于序列到序列（Sequence to Sequence，Seq2Seq）架构的摘要抽取模型。基于Transformer结构[3]构建的预训练模型BERT[14]（Bidirectional Encoder Representations from Transformers）在多个NLP任务中取得最佳表现，将其首次应用到抽取式文本摘要中[15]，在多个摘要数据集上具有较优的结果。

随着神经网络的发展，Seq2Seq的神经网络摘要模型成为研究热点。Seq2Seq模型最早用于机器翻译，将RNN作为编码器（Encoder）和解码器（Decoder）[16]，前者将输入序列表示为固定长度的向量，后者根据向量生成输出序列。文献[17]指出该模型的性能随文本长度的增加而降低。文献[18]引入注意力机制，使得解码器的每个阶段能够关注特定位置的原文，提高模型的预测能力和鲁棒性。文献[19]构建输入输出序列的模型——指针网络（Pointer network）。受神经网络机器翻译模型的启发，文献[20]首次将指针网络应用在生成式摘要任务上，在句子级摘要数据集上具有较优的性能。在此基础上，针对生成式摘要Seq2Seq模型的输出存在事实细节再现性差和容易重复的问题，文献[4]提出一种指针生成器网络，在一定程度上解决未登录词和罕见词的问题，并采用覆盖机制解决生成序列重复的问题，在CNN/Daily Mail摘要数据集上具有较优的性能。

2　数据分析

2.1　数据预处理

为促进智慧司法相关技术的发展，由政府部门、高校及企业联合举办的CAIL2020将摘要任务加入到评测比赛中，该比赛采用民事裁判文书语料，其文书和参考摘要呈现相似的段落结构，叙述顺序也基本一致，本文则是基于该语料展开研究。实验中使用的CAIL2020摘要数据集总共9 848份民事一审判决书。裁判文书预先划分成若干句子，每句都标记是否重要的标签，并提供与之对应的参考摘要全文。文书类别包括侵权责任、租赁合同、劳动合同、继承合同、借款合同等民事纠纷，文书平均包含2 568个字，最长为13 060个字，95%的文书超过4 663个字。摘要平均283个字，最长474个字，95%的摘要超过327个字。该数据集属于长文本的摘要数据集。文书和摘要的预处理分为4个步骤：1）去噪，文书和摘要包含一些空格和多余的句号等非法字符，这会影响下一步分词和分句的效果；2）信息提取，使用正则表达式对文书中标志性的字符串进行提取并识别文书类别；3）分句，按照句号划分参考摘要；4）分词，通

过导入预定义的法律名词词典和使用jieba分词工具进行分词。每句摘要中大部分信息来源于原文中标有重要标签的句子，并且摘要叙述的顺序和文书有高度的一致性，文书和摘要预处理是按照纠纷类型判定、原告诉讼请求陈述、列举案情细节及审理结果的顺序进行描述。

本文将TF-IDF向量作为句子的表示向量，利用句子向量间的余弦值衡量句子间的相似度，同时对摘要与标有重要信息标签句子的相似度进行可视化实验研究。某份文书的参考摘要和重要句子之间相似度热力图如图1所示。热力图中颜色越深表示两者相似度越高。热力图呈阶梯状，说明摘要信息分布和原文信息分布具有一致性，且重要句子和摘要句子呈现多对一的特点，即同一个重要句子大概率只与一句摘要的内容相关。

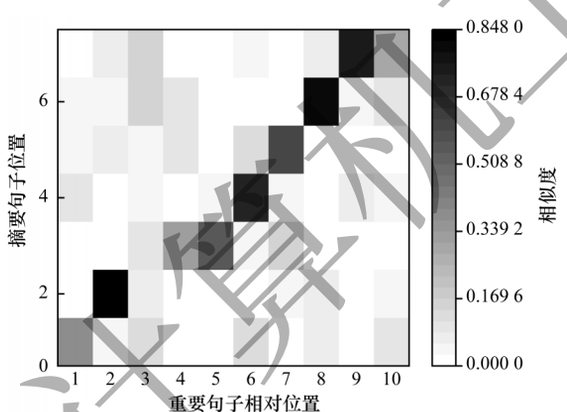

图1 参考摘要和重要句子之间相似度热力图

Fig.1 Similarity thermodynamic diagram between reference summarization and important sentences

针对长文档摘要，“抽取+生成”的流水线模型能够缩短生成模型中输入序列和输出序列的长度，从而改善模型效果[2]。本文将摘要生成过程分为抽取和生成2个阶段，采用“句子集合-摘要全文”和“句子集合-摘要句子”生成序列粒度。“句子集合-摘要全文”找出原文中重要的句子进而生成摘要全文；“句子集合-摘要句子”找出每句摘要对应的重要句子集合，通过生成模型对抽取结果进行压缩和融合，以生成单句摘要，从而缩短输出序列和输入序列的长度。

2.2 抽取式数据集构建

在CAIL2020摘要数据集中重要句子标签可用于“句子集合-摘要全文”模型，单句摘要模型所需的数据需要做进一步处理。对于L句输入文档$D=\{S_1,S_2,\cdots,S_L\}$，N句参考摘要$T=\{T_1,T_2,\cdots,T_N\}$，本文构建D中句子索引的N个集合$Z=\{Z_1,Z_2,\cdots,Z_N\}$，其中Z_i包含T_i对应D中句子的索引。文献[1]提出针对单句摘要抽取方式，通过对输入文档中N个句子进行组合得到集合，同时对这些集合进行分类，因为预设组合的数量限制，所以减少分类次数。CAIL2020摘要数据集中部分摘要对应原文句子远高于这个限制，且数量分布不均。因此，该方式处理数据集是不合适的。CAIL2020摘要数据集的特点包括摘要和对应句子位置分布的基本一致，大部分输入句子只对应一句摘要。本文给出以下假设：Z中集合均为D的子序列，且集合之间不存在重叠。本文采用类似于实体识别的标注方式对每个句子进行标注，并使用“bio”标签，“b”表示Z_i中第一个句子，“i”表示与前一个带有“i”或“b”标签构成一个集合，“o”表示不重要的句子。该方法与实体识别标注策略不同的是“b”和“i”之间、“i”和“i”之间允许“o”标签存在。

单句抽取式摘要数据集的构建通过将D中每组句子集合和对应摘要的匹配度之和作为他们的相似度，对重要句子集合分隔成摘要数目的组，使得所有组合的相似度之和最大。如已知重要句子的索引序列为{5,6,8,9,11,15,16,21}，而摘要序列为{a,b,c,d}，因此$C(7,4)=35$种方式将序列隔成4组，{{5},{6,8},{11,15,16},{21}}为其中一种方式，分别计算{5}和a、{6,8}和b、{11,15,16}和c、{21}和d的的相似度，这种方式的相似度之和最大，因此选择第5个句子的标签为“b”，第6个和第8个句子的标签设置为“b”“i”，其他非重要句子标“o”。

3 本文模型

3.1 抽取式摘要模型

抽取式摘要模型的结构如图2所示，分为句子嵌入和文档级编码器。句子嵌入采取BERT模型，文档级编码器基于多层Transformer结构，最后接入全连接层，将每个句子分成3类标签。

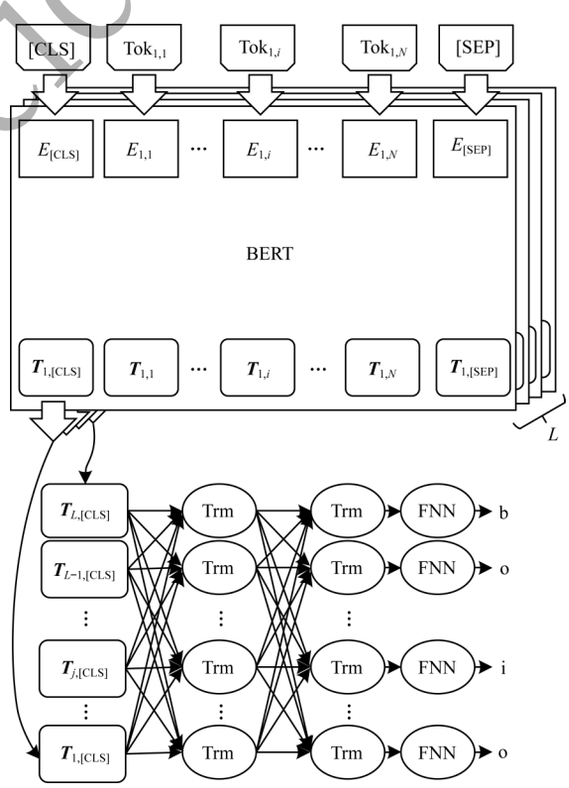

图2 抽取式摘要模型的结构

Fig.2 Structure of extractive summarization model

3.1.1　句子嵌入

BERT模型通过大量语料的自监督学习包含丰富语义信息的词表示，在文本分类任务中常使用“[CLS]”位置的输出向量作为文本表示，用于下游具体任务。在文本分类任务中，通常在每个句子前后插入“[CLS]”和“[SEP]”，然后拼接所有句子作为BERT编码器的输入。但是，CAIL2020语料中文书字数远超过BERT单次处理的序列长度，词的切分和特殊字符的引入导致输入序列增长和显卡内存不足。

本文单次处理一个句子并缓存“[CLS]”位置的输出向量$\boldsymbol{T}_{j,[\mathrm{CLS}]}$，文书第$L$个句子向量序列表示为$\boldsymbol{T}_{[\mathrm{CLS}]}=\{\boldsymbol{T}_{1,[\mathrm{CLS}]},\boldsymbol{T}_{2,[\mathrm{CLS}]},\cdots,\boldsymbol{T}_{L,[\mathrm{CLS}]}\}$。该方式能够有效减小内存，缺点是句子编码器无法进行微调。

3.1.2　句子分类器

在抽取式摘要中，句子的文档级编码方式比较常见，但此处获得句子表示是句子级编码，不包含上下文信息，只包含句子本身的信息，因此将句子向量表示输入到文档级编码器中。文档级编码器基于Transformer结构，通过全连接网络分类器对句子的文档级表示向量进行分类。在“集合-全文”的模型中使用二分类，而在“集合-句子”的分段式摘要模型中输出实体标签“b”“i”“o”3个标签，通过标签序列获得句子索引的子序列集合，例如，根据标签“{b，o，i，b，o，b}”可以得到抽取式摘要集合“{{0，2}，{3}，{5}}”。每个句子标签的概率如式(1)所示，分类指标的损失函数如式(2)所示：

$$Y=\mathrm{soft\,max}(\boldsymbol{W}_0\mathrm{Trm}(\boldsymbol{T}_{[\mathrm{CLS}]})) \tag{1}$$

$$l_{\mathrm{loss}}=-\sum_{i=1}^{L}\sum_{j=1}^{3}Z_j^i\mathrm{lb}(Y_j^i) \tag{2}$$

其中：Z^j=0，1，j={0，1，2}。

3.2　生成式摘要模型

指针生成器网络能够很好地复述事实，适用于文本摘要任务，通过计算输入序列的注意力得分，采用基于时间步的内注意力机制(Intra-Temporal Attention)[21]和解码器内的注意力机制(Intra-Decoder Attention)[22]减少输出序列的重复。

3.2.1　指针生成器网络

在编码端，将长度为L的序列$S=\{s_1,s_2,\cdots,s_L\}$的词向量输入到双向长短期记忆(Bi-directional Long Short-Term Memory，BiLSTM)网络中，将第i个位置的隐藏层向量拼接后作为编码器的输出$H=\{\boldsymbol{h}_2^{\mathrm{enc}},\boldsymbol{h}_2^{\mathrm{enc}},\cdots,\boldsymbol{h}_L^{\mathrm{enc}}\}$，如式(3)所示：

$$\boldsymbol{h}_i^{\mathrm{enc}}=[\overrightarrow{\boldsymbol{h}_i}^{\mathrm{T}},\overleftarrow{\boldsymbol{h}_i}^{\mathrm{T}}]^{\mathrm{T}} \tag{3}$$

在解码端，式(4)表示LSTM为主体构成的解码器(g函数表示解码器的计算过程)，在t时刻接收包括上一时刻隐藏层的输出$\boldsymbol{h}_{t-1}$和$\boldsymbol{c}_{t-1}$、输出结果y_{t-1}的词嵌入，以及编码器在上一时刻隐藏层向量加权和$\boldsymbol{h}_{t-1}^*$，当前时刻的隐藏层输出$\boldsymbol{h}_t$和$\boldsymbol{c}_t$。输入序列中第i个词的注意力权重值a_i^t如式(5)~式(6)所示，编码器当前时刻隐藏层的加权和$\boldsymbol{h}_t^*$如式(7)所示：

$$\boldsymbol{h}_t,\boldsymbol{c}_t=g(\boldsymbol{h}_{t-1}^*,\boldsymbol{y}_{t-1},\boldsymbol{h}_{t-1},\boldsymbol{c}_{t-1}) \tag{4}$$

$$u_i^t=\boldsymbol{V}_0\tanh(\boldsymbol{W}_1\boldsymbol{h}_i^{\mathrm{enc}}+\boldsymbol{W}_2[\boldsymbol{h}_t^{\mathrm{T}},\boldsymbol{c}_t^{\mathrm{T}}]^{\mathrm{T}}) \tag{5}$$

$$a_i^t=\mathrm{softmax}(u_i^t) \tag{6}$$

$$\boldsymbol{h}_t^*=\sum_{L}^{i=1}a_i^t\times\boldsymbol{h}_i^{\mathrm{enc}} \tag{7}$$

其中：$\boldsymbol{W}_1$、$\boldsymbol{W}_2$为参数可更新的矩阵；$\boldsymbol{V}_0$为参数可更新的行向量。

词表概率P_{vocab}和词表概率的权重p_{gen}如式(8)和式(9)所示，P_{vocab}和原文的词概率加权和得到最终拓展后的词表概率P_{exp}，如式(10)所示：

$$P_{\mathrm{vocab}}=\mathrm{soft\,max}(\boldsymbol{W}_3[(\boldsymbol{h}_t^*)^{\mathrm{T}},\boldsymbol{h}_t^{\mathrm{T}},\boldsymbol{c}_t^{\mathrm{T}},(\boldsymbol{c}_t^d)^{\mathrm{T}}]^{\mathrm{T}}) \tag{8}$$

$$p_{\mathrm{gen}}=\mathrm{sigmoid}(\boldsymbol{V}_1[\boldsymbol{h}_t^{\mathrm{T}},\boldsymbol{c}_t^{\mathrm{T}},(\boldsymbol{c}_t^d)^{\mathrm{T}},(\boldsymbol{h}_t^*)^{\mathrm{T}},(\boldsymbol{y}_{t-1})^{\mathrm{T}}]) \tag{9}$$

$$P_{\mathrm{exp}}(w)=p_{\mathrm{gen}}\times P_{\mathrm{vocab}}(w)+(1-p_{\mathrm{gen}})\times\sum_{i:w=wi}a_i^t \tag{10}$$

3.2.2　时间步内注意力机制

为解决长文本的重复问题，本文基于时间内的注意力机制，使得解码器在每个时刻关注输入序列的不同部位。注意力得分计算公式如下：

$$e_i^t=\begin{cases}\exp u_i^t, t=1\\ \dfrac{\exp u_i^t}{\sum_j^{t-1}\exp u_i^t}, t>1\end{cases} \tag{11}$$

$$a_i^t=\frac{e_i^t}{\sum_j^L e_j^t} \tag{12}$$

3.2.3　解码器内注意力机制

虽然基于时间步的注意力机制可以使解码器在不同时刻关注输入序列的不同部位，但是如果解码器自身的隐藏层在上一时刻的输出和此时刻相似，存在生成重复序列的可能。因此，本文通过解码器所有隐藏层输出向量计算注意力，并求加权和作为t时刻输入的方式，其计算过程如下：

$$e_{tt'}^d=(\boldsymbol{V}'(\boldsymbol{W}'\boldsymbol{h}_{t'}^d+\boldsymbol{W}\boldsymbol{h}_t^d)) \tag{13}$$

$$a_{tt'}^d=\frac{\exp(e_{tt'}^d)}{\sum_{j=1}^{t-1}\exp(e_{tj}^d)} \tag{14}$$

$$\boldsymbol{c}_t^d=\sum_{j=1}^{t-1}a_{tj}^d\boldsymbol{h}_j^d \tag{15}$$

其中：$\boldsymbol{W}$’和$\boldsymbol{W}$为可更新参数的矩阵；$\boldsymbol{V}$’为可更新参数的行向量。

3.2.4　损失函数

在训练阶段，参考摘要在该时刻出现的字为w_t，t时刻输出的拓展词表概率为$P_{\mathrm{exp}}(w_t)$，损失函数如式(16)所示，对所有时刻的损失函数求平均值得到最终的损失函数，如式(17)所示：

$$l_t^{\text{loss}} = -\text{lb}(P_{\text{exp}}(w^t)) \tag{16}$$

$$l_{\text{loss}} = \frac{1}{T}\sum_{t=1}^{T} l_t^{\text{loss}} \tag{17}$$

4 实验与结果分析

4.1 实验设置

在抽取式摘要模型中BERT模型采用追一科技有限公司推出的以词为单位的预训练模型WoBERT，使用d层Transformer结构，其中全连接层的隐藏层为2 048，注意力层为8，输出全连接层的隐藏层为128。RNN编码器-解码器的隐藏层均为256，以字为单位构建词表的大小为4 847，采用随机初始化方式进行字嵌入。在“集合-全文”语料上最大编码长度设置为1 370，最小解码长度设置为330，“集合-句子”语料最大编码长度和最小解码长度分别设置为360、120，均能满足95%左右的输入序列和输出序列的长度要求，采用束搜索方式进行预测，Beam Size设置为3。所使用的优化器均为Adam优化器[23]。抽取式摘要模型和生成式摘要模型的参数设置如表1所示。

表1 抽取式摘要模型和生成式摘要模型的参数设置

Table 1 Parameter settings of extractive summarization model and generative summarization model

模型	批尺寸	学习率	训练步数	累计梯度步数
抽取式摘要模型	32	0.000 005	40	4
生成式摘要模型	8	0.000 040	20	4

训练集和测试集按照文书类别等比例划分成8∶2，并在抽取式摘要模型的训练阶段中使用训练集的10%用于验证。由于生成式摘要模型的预测比较耗时，因此没有在训练阶段进行验证。

4.2 结果分析

4.2.1 评价指标

大多数分类系统通常使用精确率(Precision)、召回率(Recall)及F1值(F1-score)作为评价指标，如式(18)~式(20)所示：

$$P = \frac{T_{\text{TP}}}{T_{\text{TP}} + F_{\text{FP}}} \tag{18}$$

$$R = \frac{T_{\text{TP}}}{T_{\text{TP}} + F_{\text{FN}}} \tag{19}$$

$$\text{F1} = \frac{2 \times P \times R}{P + R} \tag{20}$$

真实正例的集合用S表示，预测为正例的集合用S^*表示。预测正确的正例表示S和S^*中同时存在的元素，预测错误的负例表示S中不存在而S^*中存在的元素，预测错误的正例表示S中存在而S^*中不存在的元素，它们的数量分别用T_{TP}、F_{FP}、F_{FN}表示。

本文以参考摘要作为基准评价系统摘要质量的评价指标ROUGE[24](Recall-Oriented Understudy for Gisting Evaluation)，其是一种常用的文本摘要评价指标。ROUGE-N的召回率如式(21)所示：

$$R_{\text{ROUGE}-N} = \frac{\sum_{S \in \{\text{RefSums}\}} \sum_{\text{gram}_n \in S} \text{Count}_{\text{match}}(\text{gram}_N)}{\sum_{S \in \{\text{RefSums}\}} \sum_{\text{gram}_n \in S} \text{Count}(\text{gram}_N)} \tag{21}$$

其中：gram_N表示文本中长度为N的连续子串；分母为所有参考摘要的所有N-gram的数量。PredSums表示模型预测的摘要，RefSums表示预测摘要与参考摘要共现的N-gram的数量。精确率的计算如式(22)，ROUGE-L的计算如式(23)~式(24)所示，LCS函数表示2个序列最长公共子串的长度，len函数为序列总长度，2类指标的F1值如式(25)所示：

$$P_{\text{ROUGE}-N} = \frac{\sum_{S \in \{\text{PredSums}\}} \sum_{\text{gram}_n \in S} \text{Count}_{\text{match}}(\text{gram}_N)}{\sum_{S \in \{\text{PredSums}\}} \sum_{\text{gram}_n \in S} \text{Count}(\text{gram}_N)} \tag{22}$$

$$R_{\text{ROUGE}-L} = \frac{\text{LCS}(\text{RefSums}, \text{PredSums})}{\text{len}(\text{RefSums})} \tag{23}$$

$$P_{\text{ROUGE}-L} = \frac{\text{LCS}(\text{RefSums}, \text{PredSums})}{\text{len}(\text{PredSums})} \tag{24}$$

$$\text{F1} = \frac{2 \times R \times P}{R + P} \tag{25}$$

为探究抽取式摘要模型对信息识别准确性及生成结果的影响，本文分别用Z和Z'表示真实集合和预测集合。假设真实抽取结果为{{0}，{3,5}，{10,12,13}}，预测结果为{{0}，{4,5}，{10,12,13}}。评价指标主要有2个：1)exact_match，评价模型识别完整的句子集合的能力，例如，{0}和{10,12,13}存在于Z和Z'中，TP等于2；2)all，评价模型识别单个句子的能力，分别取Z和Z'中所有集合的并集来替换真实和预测集合，例如，Z={0,3,5,10,12,13}，Z'={0,4,5,10,12,13}，TP等于5。

4.2.2 生成式摘要模型的结果分析

本文采用2.2节的方法处理数据，将原来重要句子的集合分为若干组，其中有2例摘要句子数目少于标有重要标签的数目，因此他们无法处理而舍弃。处理前后语料的ROUGE指标对比如表2所示。

表2 处理前后语料的ROUGE指标对比

Table 2 ROUGE indexs comparison of the corpus before and after processing %

语料	ROUGE-1			ROUGE-2			ROUGE-L		
	P	R	F1	P	R	F1	P	R	F1
处理前语料	36.31	91.47	50.94	28.25	69.89	39.43	40.93	84.88	54.59
处理后语料	37.66	82.09	47.74	28.95	56.31	35.82	40.82	80.61	50.89

处理前语料是句子集合拼接的序列和摘要全文的ROUGE指标，处理后语料是计算单句摘要和对应一组句子集合拼接而成序列的指标。处理后语料的召回率比处理前下降比较多，这是由于有些句子划分到错误的一组中或者摘要和文书没按照同一顺序叙述。

为对比序列长度对生成模型的影响，在“句子集合-摘要全文”“句子集合-摘要句子”语料上生成式摘要模型的ROUGE指标如表3所示。

表3　在“集合-全文”和“集合-句子”语料上生成式摘要模型的ROUGE指标

Table 3　ROUGE indexs of generative summarization model on "collection-full text" and "collection-sentence" corpus　%

语料	ROUGE-1			ROUGE-2			ROUGE-*L*		
	P	*R*	F1	*P*	*R*	F1	*P*	*R*	F1
“集合-全文”	66.38	75.37	70.32	53.86	61.26	57.10	64.56	72.98	68.28
“集合-句子”	71.35	76.80	72.74	60.23	64.56	61.29	70.21	77.08	72.82

本文将在“集合-句子”语料上生成来源于同一文书的摘要句子拼在一起后与“集合-全文”上的结果进行对比。从表3可以看出，在“集合-句子”语料上训练模型的指标均优于“集合-全文”语料，而且在“集合-句子”语料中有一部分参考摘要和原文句子的组合不匹配，因此，在CAIL2020数据集上该模型能够有效缩短生成模型的输入输出序列长度。

4.2.3　流水线模型结果分析

抽取式摘要模型的分类指标如表4所示，不同模型的输出摘要与参考摘要间的ROUGE指标如表5所示。

表4　抽取式摘要模型的评价指标

Table 4　Evaluation indexs of extractive summarization models　%

模型	exact_match			all		
	P	*R*	F1	*P*	*R*	F1
Lead-3模型	—	—	—	2.16	0.50	0.81
BERT+Trm^4(3)+crf模型	46.96	39.25	42.76	87.89	81.04	84.33
BERT+ Trm^2(3) 模型	49.67	41.80	45.40	91.49	75.31	82.62
BERT+ Trm^4(3) 模型	50.74	40.19	44.85	89.25	80.66	84.74
BERT+ Trm^4(2) 模型	—	—	—	87.91	83.24	85.51

表5　不同模型的ROUGE指标

Table 5　ROUGE indexs of different models　%

模型	ROUGE-1			ROUGE-2			ROUGE-*L*		
	P	*R*	F1	*P*	*R*	F1	*P*	*R*	F1
Lead-3模型	8.21	5.39	1.00	1.08	0.06	0.12	8.84	1.12	1.97
BERT+Trm^4(3)+crf+PG^{1}模型	71.17	68.91	68.68	58.19	56.11	56.04	67.70	69.08	67.62
BERT+ Trm^2(3)+ PG^{1}模型	71.37	67.17	67.48	58.49	54.92	55.22	68.16	67.45	66.82
BERT+ Trm^4(3)+ PG^{1}模型	72.30	66.22	67.65	59.29	54.19	55.41	68.61	67.14	67.02
BERT+ Trm^4(2)+ PG^{0}模型	64.50	72.92	68.13	51.11	57.87	54.00	61.83	69.35	65.06

本文baseline模型是Lead-3，即取输入文档的前3句作为摘要。“BERT+Trm^4(3)”表示使用BERT作为句子嵌入和4层Transformer结构，“3”和“2”分别代表三分类（“bio”标签的形式）和二分类（即是否标有“重要”标签）。“crf”表示引入条件随机场（Conditional Random Field，CRF）[25]。“PG”表示将抽取的结果输入到指针生成器网络中，上标“0”和“1”表示在“集合-全文”和“集合-句子”的语料训练的模型。

在该数据集上Lead-3模型的性能较差，其原因为相比新闻文档，法律文书的关键信息一般在末尾或者分布比较均匀。在“exact_match”指标下，在Lead-3模型上增加CRF，其精确率、召回率及F1下降。在“all”指标下，“BERT+Trm^4(2)”的召回率最高，其次是“BERT+Trm^4(3)+crf”。BERT+Trm^4(2)+PG^{0}模型抽取的句子最全，其ROUGE-1、ROUGE-2、ROUGE-*L*的平均值为62.40%，而BERT+Trm^4(3)+crf+PG^{1}模型ROUGE-1、ROUGE-2、ROUGE-*L*的平均值为64.11%，比前者提高了1.69个百分点。因此，在抽取过程中ROUGE指标的召回率下降表示在内容过程中部分信息被遗漏，提高召回率，使得流水线模型的总体表现提高。在生成的过程中舍弃部分冗余或者不重要的信息，以提高精确率。

5　结束语

本文提出一种基于序列标注的分段式摘要模型。通过将CAIL2020转换成分段式的摘要抽取数据集，将“bio”实体识别标签用于句子抽取和段落划分，缩短生成模型的输入序列长度。实验结果表明，该模型在处理生成句子上ROUGE-1、ROUGE-2和ROUGE-*L*的F1平均值为64.11%。后续将基于流水线式摘要的架构，不依赖语料本身的结构特点，围绕抽取句子分组问题开展研究。

参考文献

[1] LEBANOFF L, SONG K, DERNONCOURT F, et al. Scoring sentence singletons and pairs for abstractive summarization[EB/OL]. [2021-02-15]. https://arxiv.org/abs/1906.00077.

[2] SUBRAMANIAN S, LI R, PILAULT J, et al. On extractive and abstractive neural document summarization with

transformer language models[EB/OL]. [2021-02-15]. https://arxiv. org/abs/1909. 03186.
[3] VASWANI A, SHAZEER N, PARMAR N, et al. Attention is all you need[C]//Proceedings of the 31st International Conference on Neural Information Processing Systems. NewYork, USA: ACM Press, 2017: 6000-6010.
[4] SEE A, LIU P J, MANNING C D. Get to the point: summarization with pointer-generator networks[EB/OL]. [2021-02-15]. https://arxiv. org/pdf/1704. 04368. pdf.
[5] LUHN H P. The automatic creation of literature abstracts[J]. IBM Journal of Research and Development, 1958, 2(2): 159-165.
[6] EDMUNDSON H P. New methods in automatic extracting[J]. Journal of the ACM, 1969, 16(2): 264-285.
[7] LIN C Y, HOVY E. Identifying topics by position[EB/OL]. [2021-02-15]. http://citeseerx. ist. psu. edu/viewdoc/download;jsessionid=18A483239494376ACA4520EFBD7 42575?doi=10. 1. 1. 13. 8985&rep=rep1&type=pdf.
[8] KUPIEC J, PEDERSEN J, CHEN F. A trainable document summarizer [C]//Proceedings of the 18th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval. New York, USA: ACM Press, 1995: 68-73.
[9] LIN C Y. Training a selection function for extraction[C]// Proceedings of the 18th International Conference on Information and Knowledge Management. New York, USA: ACM Press, 1999: 55-62.
[10] MIHALCEA R, TARAU P. Textrank: bringing order into text[EB/OL]. [2021-02-15]. https://www. researchgate. net/profile/Paul-Tarau/publication/200042361_TextRank_Bringing_Order_into_Text/links/0912f508a98af2fe2400 0000/TextRank-Bringing-Order-into-Text. pdf.
[11] ERKAN G, RADEV D R. LexRank: graph-based lexical centrality as salience in text summarization[EB/OL]. [2021-02-15]. https://arxiv. org/abs/1109. 2128v1.
[12] NALLAPATI R, ZHAI F, ZHOU B. SummaRuNNer: a recurrent neural network based sequence model for extractive summarization of documents[C]//Proceedings of the 31st AAAI Conference on Artificial Intelligence. [S. l.]: AAAI Press, 2017: 3075-3081.
[13] ZHOU Q, YANG N, WEI F, et al. Neural document summarization by jointly learning to score and select sentences[EB/OL]. [2021-02-15]. https://arxiv. org/pdf/1807. 02305. pdf.
[14] DEVLIN J, CHANG M W, LEE K, et al. BERT: pre-training of deep bidirectional transformers for language understanding[EB/OL]. [2021-02-15]. https://arxiv. org/pdf/1810. 04805. pdf.
[15] LIU Y, LAPATA M. Text summarization with pretrainedencoders [EB/OL]. [2021-02-15]. https://arxiv. org/abs/1908. 08345.
[16] SUTSKEVER I, VINYALS O, LE Q V. Sequence to sequence learning with neural networks[J]. Advances in Neural Information Processing Systems, 2014, 4(1): 3104-3112.
[17] CHO K, VAN MERRIËNBOER B, GULCEHRE C, et al. Learning phrase representations using RNN encoder-decoder for statistical machine translation[C]//Proceedings of Conference on Empirical Methods in Natural Language Processing. [S. l.]: AAAI Press, 2014: 1724-1734.
[18] BAHDANAU D, CHO K H, BENGIO Y. Neural machine translation by jointly learning to align and translate[EB/OL]. [2021-02-15]. https://arxiv. org/pdf/1409. 0473. pdf.
[19] VINYALS O, FORTUNATO M, JAITLY N. Pointer networks[EB/OL]. [2021-02-15]. https://arxiv. org/pdf/1506. 03134. pdf.
[20] RUSH A M, CHOPRA S, WESTON J. A neural attention model for abstractive sentence summarization[EB/OL]. [2021-02-15]. https://arxiv. org/pdf/1509. 00685. pdf.
[21] SANKARAN B, MI H, AL-ONAIZAN Y, et al. Temporal attention model for neural machine translation[EB/OL]. [2021-02-15]. https://arxiv. org/abs/1608. 02927.
[22] PAULUS R, XIONG C, SOCHER R. A deep reinforced model for abstractive summarization[EB/OL]. [2021-02-15]. https://arxiv. org/pdf/1705. 04304. pdf.
[23] KINGMA D P, BA J L. Adam: a method for stochastic optimization[EB/OL]. [2021-02-15]. http://de. arxiv. org/pdf/1412. 6980.
[24] LIN C Y. Rouge: a package for automatic evaluation of summaries[EB/OL]. [2021-02-15]. http://citeseerx. ist. psu. edu/viewdoc/download; jsessionid=657245BA3216 0BCB52B6B8972D0FE238? doi=10. 1. 1. 126. 4764&rep=rep1&type=pdf.
[25] LAFFERTY J. , MCCALLUM A, PEREIRA F. Conditional random fields: Probabilistic models for segmenting and labeling sequence data [C]//Proceedings of the 18th International Conference on Machine Learning. New York, USA: ACM Press, 2001: 282-289.

编辑 薛晋栋