第3章

基于Transformer模型的NER

3.1 引言

随着数字化技术的快速发展，越来越多的古籍文献可以使用数字化存储和传播。然而，古籍文本的复杂性和特殊性给其信息抽取和文本分析带来了巨大挑战。在古籍研究领域，准确地识别和标注文本中的命名实体对于深入挖掘古籍的知识、历史和文化具有重要意义。

在古籍识别任务中，NER 的目标是自动识别文本中的人名、地名、机构名等实体，以及日期、时间等具有特定意义的实体。这些实体信息是深入理解古籍文本、还原历史场景、揭示人物关系和地理背景的重要线索。

基于 Transformer 模型的 NER 技术作为 NLP 领域的重要突破，为古籍文本识别提供了新的解决方案。本章旨在探索基于 Transformer 模型的 NER 方法在古籍识别中的应用，本章将使用在 Transformer 架构基础上引申出来的基于分层 Transformer 模型的 NER、基于 BERT-CRF（Bidirectional Encoder Representations from Transformers with Conditional Random Fields，基于条件随机场转换器的双向编码表示）的 NER 和基于迁移学习的细粒度 BERT（Bidirectional Encoder Representations from Transformers，基于转换器的双向编码表示）的 NER 三种方法，用于古籍实体识别的研究。

3.2 问题引入

古籍文献作为珍贵的文化遗产，蕴含着丰富的历史、文学和社会信息，对于古代社会、文化传承和学术领域都具有重要价值。然而，由于古籍文本的特殊性和复杂性，对其进行深入的分析和理解一直面临着挑战。在此背景下，如何利用先进的 NLP 技术来提升古籍文献的识别和分析能力成为一个重要的问题。

NER 是 NLP 中的一个关键任务，旨在识别文本中具有特定语义含义的命名实

体，如人名、地名、时间、收藏地等。在古籍文本实体识别任务中，准确地识别和标注古籍文本中的命名实体是一项具有挑战性的任务。古籍文本常常使用古代汉字、异体字和古代人名、地名的表示形式，使得传统的基于规则或词典的方法往往无法满足准确实体识别的要求。

Transformer 模型能够通过学习上下文信息，对复杂的古籍文本进行更准确的 NER。然而，将基于 Transformer 模型的 NER 方法应用于古籍文献识别任务中仍然存在一些问题和挑战。首先，古籍文献的特殊性导致其数据量有限，相比于现代语料库，可用于训练的标注数据较少。如何在有限的古籍数据上训练出高性能的 NER 模型，成为一个亟待解决的问题。其次，古籍文本中存在着各种异体字、俗字和缺失字等，这些问题对于命名实体的准确识别带来了挑战。此外，古籍文献中的人名、地名常常具有多义性和上下文依赖性，如何准确区分不同的命名实体，尤其是在上下文信息有限的情况下，也是一个需要解决的问题。基于上述问题，本章旨在探索基于 Transformer 模型的 NER 方法在古籍命名实体识别中的应用，利用 Transformer 模型提升古籍文献的 NER 性能。

3.3 基于分层 Transformer 模型的 NER

3.3.1 引言

NLP 的根本目标之一是开发能够理解人类语言所表达的潜在语义系统。实现这一目标的一个重要步骤是能够有效地提取有用的语义信息，例如，从给定的文本中识别出命名实体的边界以及命名实体的类型，也就是 NER，它是信息抽取中的标准任务之一。

在大规模地应用深度学习之前，NER 方法主要以基于词典和规则的方法、传统机器学习的方法为主，识别准确率在缓慢提升，但一直无法取得理想效果。深度学习的引入和发展使 NER 任务的准确率得到了很大的提高。开发能够高效识别嵌套命名实体的算法对于许多 NLP 下游任务的执行至关重要，而且具有一定的实际应用价值，如 RE、EE、共参分解、知识库问答、信息检索和语义解析等。然而，嵌套命名实体结构是复杂多变的，嵌套颗粒度与嵌套层数缺乏规律性，如何快速高效地从开放领域的文本中准确获取嵌套命名实体结构信息，使得语义理解更加精准，是 NLP 进入全面化应用的关键。

早期，人们常结合基于规则和基于机器学习的方法来处理嵌套命名实体。文献首先使用隐马尔可夫模型识别最内层的非嵌套命名实体，其次使用基于规则的后处理来识别外部命名实体，最后在 GENIA 数据集上评估。Alex 等人于 2017 年提出了几种基于 CRF 的技术，用于对 GENIA 数据集进行嵌套 NER。该方法以特定的顺序对实

体类型应用 CRF，这样每个 CRF 都能利用前面 CRF 的输出，采用这种级联方法可以取得最佳嵌套 NER 结果。2009 年，Finkel 和 Manning 从解析的角度来实现嵌套 NER 任务，通过构建选区树，将命名实体映射到树中的节点上。

3.3.2　实现原理与步骤

基于分层 Transformer 模型的 NER 的实现原理主要涉及两方面，分别是分层 Transformer 模型和标注数据的预处理。

1）分层 Transformer 模型

分层 Transformer 模型是一种基于 Transformer 的模型架构，用于处理 NLP 任务。它由多层 Transformer 编码器和解码器组成，其中编码器负责将输入文本表示为上下文有关的向量表示，解码器负责生成输出结果。

在 NER 任务中，输入是一个句子，目标是识别出句子中的命名实体，如人名、地名、组织机构等。为了实现这一目标，分层 Transformer 模型首先将输入句子的每个单词转换为其对应的词向量。然后，通过多个编码器层，逐步将输入句子的上下文信息编码到词向量中。每个编码器层包括多头自注意力机制和前馈神经网络层。多头自注意力机制可以捕获单词之间的依赖关系，前馈神经网络层用于对词向量进行非线性变换。通过多个编码器层的堆叠，模型可以逐渐提取句子中的上下文信息。

在编码器的输出上，可以应用 CRF 层来对每个单词进行标签预测。CRF 层考虑了相邻单词之间的标签依赖关系，可以提高模型在命名实体边界的准确性。最终，模型会输出每个单词的命名实体标签。

2）标注数据的预处理

在训练分层 Transformer 模型之前，需要对标注数据进行预处理。通常，标注数据由句子和每个单词对应的命名实体标签组成。预处理包括将句子转换为词向量表示以及将命名实体标签转换为模型可接受的形式。句子的转换可以使用预训练的词向量模型，例如 Word2Vec 或 GloVe，将每个单词映射为词向量。这些词向量可以在分层 Transformer 模型的输入层使用。命名实体标签的转换通常采用 BIO（Begin，Inside，Outside）或者 BIOES（Begin，Inside，Outside，End，Single）格式。BIO 和 BIOES 是常用的命名实体标签编码方案，用于表示命名实体的开始、内部、结束和单独的状态。通过预处理标注数据，可以将其转换为模型可接受的输入形式，从而训练分层 Transformer 模型进行 NER。

总之，基于分层 Transformer 的 NER 的实现原理包括分层 Transformer 模型的架构和标注数据的预处理。这些技术可以帮助模型从输入句子中提取上下文信息，并预测每个单词的命名实体标签。使用分层 Transformer 模型进行 NER，步骤如下。

(1) 数据准备。

收集或准备一个 NER 标注数据集,其中包含句子和每个单词对应的标签(如人名、地名、组织机构等)。并将标注数据集划分为训练集、验证集和测试集。

(2) 模型构建。

选择一个适合的分层 Transformer 模型,如 BERT 或 GPT(Generative Pre-trained Transformer,生成式预训练变换器)。根据任务需求,可以选择在预训练的模型上进行微调(fine-tuning),或从头开始训练。在模型中添加适当的层和参数,以输出每个单词的命名实体标签。

(3) 输入数据处理。

将句子转换为词向量表示,可以使用预训练的词向量模型,如 Word2Vec 或 GloVe,将每个单词映射为词向量。将标签转换为模型可接受的形式,通常使用 BIO 或 BIOES 编码方案。

(4) 模型训练。

使用训练集进行模型的训练。将输入数据(词向量表示)和标签输入模型中,计算预测结果,并与真实标签进行比较,计算损失函数。根据损失函数的反向传播算法,更新模型的参数,以减小预测结果与真实标签之间的差距。重复迭代训练过程,直到模型收敛或达到指定的训练轮数。

(5) 模型评估。

使用验证集对模型进行评估,计算准确率、召回率、$F1$ 评分等指标,以评估模型的性能。根据评估结果,可以对模型进行调整和优化。

(6) 模型应用。

使用测试集对模型进行最终评估,评估模型在未使用过的数据集上的性能。将训练好的模型部署到实际应用中,对新的文本进行 NER。需要注意的是,在实际应用中,可以根据具体需求对模型进行调整和优化,例如增加正则化、模型融合、后处理等技术,以提高 NER 的性能。

3.3.3 基本结构与训练方法

1. 基本结构

嵌套 NER 框架如图 3.1 所示,该框架主要包含 5 部分内容。

(1) 获取有标签数据集:人工标注获取有标签嵌套命名的实体数据集。常见的标注方法如下。

① BIO 标注法,其中 B 表示开始,I 表示内部,O 表示外部。

② BIOES 标注法,该标注方法是 BIO 标注法的延伸,其中 E 表示结束,S 表示单

独构成一个命名实体。

(2) 构建词语向量表示：对原始的输入字符序列进行分词，将分词表示成计算机可识别的计算类型，在各个位置上学习一个位置向量表示来编码序列顺序的信息，或者在表示以外加入一些传统的浅层有监督模型中使用的特征。

(3) 进行特征提取：对词向量表示进行特征变换、编码，例如使用递归神经网络、CNN、Transformer 等进行建模和学习。

(4) 嵌套 NER：采用一定的模型对词语向量表示和提取的特征进行训练，得到嵌套 NER 预训练模型。

(5) 评估识别性能：对识别结果进行评测，评测流程见图 3.1。

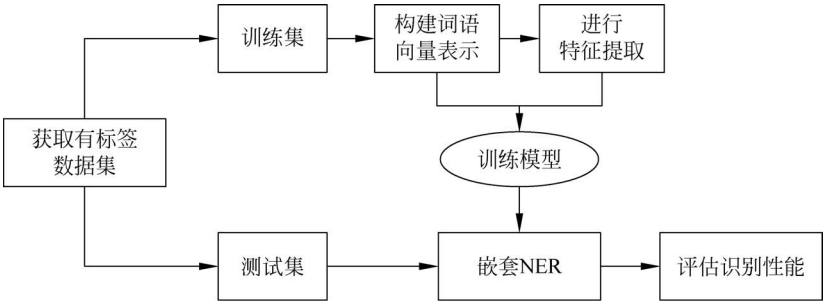

图 3.1　评测流程

分层 Transformer 模型是一种基于 Transformer 的模型架构，用于处理 NLP 任务。它主要由多个编码器层组成，每个编码器层由多头自注意力机制和前馈神经网络层组成。以下是分层 Transformer 模型的基本结构。

(1) 输入嵌入层(Input Embedding Layer)。

将输入的文本序列中的每个单词转换为其对应的词向量表示。可以使用预训练的词向量模型，如 Word2Vec 或 GloVe，或者在模型训练时从头开始学习词向量。

(2) 编码器层(Encoder Layer)。

多个编码器层的堆叠是分层 Transformer 模型的核心。每个编码器层可以独立地对输入序列进行处理，并且每个编码器层的输出将作为下一个编码器层的输入。每个编码器层包含两个子层：多头自注意力机制层和前馈神经网络层。

(3) 多头自注意力机制层(Multi-Head Self-Attention Layer)。

在多头自注意力机制层中，输入序列的每个单词都会与其他单词进行交互，以捕捉单词之间的依赖关系。通过使用多个注意力头，模型可以同时学习多个注意力权重，以捕捉不同的语义信息。注意力权重告诉模型每个单词与其他单词的关联程度。

(4) 前馈神经网络层(Feed-Forward Neural Network Layer)。

在前馈神经网络层中，通过对多头自注意力机制的输出进行非线性变换。通常，前馈神经网络层由两个全连接层和一个激活函数组成，用于引入非线性。

(5) 输出层(Output Layer)。

在 NER 任务中,可以在编码器的输出上应用 CRF 层。CRF 层考虑了相邻单词之间的标签依赖关系,可以提高模型在命名实体边界的准确性。最终,模型会输出每个单词的命名实体标签。通过多个编码器层的堆叠,分层 Transformer 模型可以逐渐提取输入序列的上下文信息,并预测每个单词的命名实体标签。

需要注意的是,分层 Transformer 模型可以根据具体任务的需求进行调整和扩展。例如,在 NER 中,可以在模型中添加 CRF 层以考虑标签依赖关系,或者使用模型融合等技术来提高性能。

2. 训练方法

分层 Transformer 的训练方法通常采用两个阶段:预训练阶级和微调阶级。

(1) 预训练阶段。

在预训练阶段,使用大规模的未标注文本数据集对分层 Transformer 模型进行训练。通常使用自监督学习的方法,如掩码语言建模(Masked Language Modeling,MLM)或预测下一个句子(Next Sentence Prediction,NSP)来训练模型。在 MLM 任务中,一部分输入文本中的单词会被随机掩码,模型需要预测这些掩码单词的正确标签。在 NSP 任务中,模型需要判断两个句子是否相邻,以帮助模型学习句子级别的关系。

(2) 微调阶段。

在预训练完成后,使用有标签的任务特定数据集对模型进行微调,以适应具体的任务。在 NER 任务中,可以准备一个 NER 标注的数据集,其中包含句子和每个单词对应的命名实体标签。将 NER 数据集输入模型中,计算预测结果,并与真实标签进行比较,计算损失函数。根据损失函数的反向传播算法来更新模型的参数,以减小预测结果与真实标签之间的差距。重复迭代训练过程,直到模型收敛或达到指定的训练轮数。在微调阶段,可采用下述策略来进一步优化模型性能。

① 学习率调整:可以使用学习率衰减策略,如逐渐减小学习率或在训练过程中进行学习率的动态调整。

② 正则化:可以使用正则化技术,如权重衰减(Weight Decay)或随机删除神经元(Dropout),以防止过拟合。

③ 模型融合:可以将多个训练好的模型进行融合,如使用集成学习方法或模型堆叠方法,以提高泛化能力。需要注意的是,分层 Transformer 的预训练和微调方法可以根据具体的任务需求进行调整和优化。此外,采用更大规模的预训练数据集和更复杂的训练策略通常可以提高模型的性能。

3. 预训练模型

分层 Transformer 模型的预训练模型有许多，其中一些最为知名和常用的预训练模型包括以下几种。

(1) BERT。

BERT 是一种基于 Transformer 架构的预训练模型，通过 MLM 和 NSP 的任务来进行训练。BERT 模型在多种 NLP 任务上取得了显著的性能提升，并广泛应用于文本分类、NER 等任务中。

(2) GPT。

GPT 是一种基于 Transformer 架构的预训练模型，通过语言模型任务进行训练。GPT 模型使用自回归的方式生成下一个单词，以捕捉句子中的上下文关系，并能够生成连贯的句子。

(3) RoBERTa。

RoBERTa 是在 BERT 模型的基础上进行改进和优化的预训练模型。RoBERTa 采用更大的模型规模和更长的预训练步骤，以获得更好的性能表现。

(4) ALBERT。

ALBERT 是对 BERT 模型进行改进，以减小模型规模和参数数量的预训练模型。ALBERT 通过共享参数和分解参数矩阵等技术，实现模型轻量化，提高了训练和推断的效率。

除了以上列举的模型，还有许多其他的分层 Transformer 预训练模型，如 XLNet、T5、ELECTRA 等。这些模型在不同的任务和数据集上表现出色，具有不同的优势和特点。需要根据具体的任务需求、数据集和计算资源等因素选择适合的预训练模型，并结合微调和优化技术来进一步提高模型的性能。

4. 损失函数

分层 Transformer 模型的损失函数可以根据具体任务进行选择。以下是几种常见的损失函数示例。

(1) MLM。

在预训练阶段，分层 Transformer 模型通常使用 MLM 任务进行训练。在该任务中，输入文本中的一部分单词会被随机掩码，模型需要预测这些掩码单词的正确标签。通常，采用交叉熵损失函数来计算模型预测结果与真实标签之间的差异。

(2) NSP。

在预训练阶段，分层 Transformer 模型还可以通过使用预测下一个句子的方式来进行训练。在该任务中，模型需要预测两个句子是否是相邻的。通常，采用二元交叉

熵损失函数来计算模型预测的概率分布与真实标签概率分布之间的差异，具体来说，即计算相邻或不相邻句子的模型预测概率与真实标签之间的差异。

(3) NER。

在微调阶段，针对 NER 任务，常用的损失函数是序列标注任务中的交叉熵损失函数。模型将根据输入句子对每个单词进行命名实体标签的预测，并与真实标签进行比较。交叉熵损失函数可度量模型预测结果与真实标签之间的差异，以便通过反向传播算法来更新模型参数。

除上述示例损失函数外，还可以根据具体任务的特点和需求定义其他适合的损失函数。例如，在机器翻译任务中，可以使用序列级别的交叉熵损失函数来计算预测结果与目标序列之间的差异。需要根据具体任务的特点选择合适的损失函数，并结合其他优化技术来进一步提高分层 Transformer 模型的性能和泛化能力。

5. 评估指标

评估分层 Transformer 模型的性能时，可以使用多种指标来衡量其在不同任务上的表现。以下是一些常用的评估指标示例。

(1) 准确率(Accuracy)。准确率是最常见的评估指标，用于评估分类任务的性能。它表示正确分类模型在所有样本中的比例。

(2) 精确率(Precision)和召回率(Recall)。精确率和召回率通常一起使用，用于评估二分类和多分类任务中的模型性能。精确率衡量模型预测为正类别的样本中，实际为正类别的比例。召回率衡量实际为正类别的样本中，模型预测为正类别的比例。

(3) $F1$ 评分($F1$-Score)。$F1$ 评分综合了精确率和召回率，用于评估二分类和多分类任务中的模型性能。$F1$ 评分是精确率和召回率的调和平均值，能够综合考虑模型的预测准确性和对正例的召回能力。

(4) 均方误差(Mean Squared Error，MSE)。MSE 常用于回归任务中，用于评估模型预测结果与真实值之间的差异。它计算预测值与真实值之间差异的平方的平均值。

(5) 对数损失(Log Loss)。对数损失常用于概率预测任务中，用于衡量模型对概率分布的预测准确性。它度量模型预测结果与真实结果之间的差异的平均对数概率。

(6) 特定于任务的评估指标。在一些特定的任务中，可能还会使用任务特定的评估指标。例如，在 NER 任务中，可以使用标签级别的精确率、召回率和 $F1$ 评分来评估模型对命名实体的识别能力。在评估分层 Transformer 模型性能时，应根据任务类型和需求选择合适的评估指标。同时，交叉验证和对比实验等方法有助于更全面和准确地评估模型在各项指标上的性能。

3.3.4　示例

这里以 Python 代码的形式给出一个基于 TensorFlow 实现的分层 Transformer NER 模型示例。

```
import tensorflow as tf
from tensorflow.keras.layers import Dense, Input
from tensorflow.keras.models import Model
# 输入表示层
input_ids = Input(shape=(MAX_SEQ_LEN,), dtype=tf.int32)
input_masks = Input(shape=(MAX_SEQ_LEN,), dtype=tf.int32)
segment_ids = Input(shape=(MAX_SEQ_LEN,), dtype=tf.int32)
# 分层 Transformer 编码器
x = Embedding(vocab_size, d_model)(input_ids) # 词嵌入
x = AddPositionEncoding(x)                    # 添加位置编码
for n in range(n_layers):
x = MultiHeadAttention(d_model, n_heads)([x, x, x, input_masks]) # 多头自注意力
x = FeedForward(d_model)([x])                 # 前馈神经网络
# 序列标记层
x = Dense(n_tags)(x)
output = CRF(n_tags)(x)                       # CRF 层
# 构建模型
model = Model(inputs=[input_ids, input_masks, segment_ids], outputs=output)
# 编译和训练模型
model.compile(optimizer=Adam(learning_rate), loss=crf_loss)
model.fit([input_ids, input_masks, segment_ids], labels, epochs=n_epochs)
# 预测和后处理
pred = model.predict([input_ids, input_masks, segment_ids])
pred = post_process(pred)                     # 后处理,如去除冗余实体、合并实体等
```

该模型主要包含以下几部分。

(1) 输入表示层：使用词嵌入和位置编码将词序列转换为向量序列。

(2) 分层 Transformer 编码器：包含多层 Transformer 编码器，每个编码器层包括多头自注意力和前馈神经网络。

(3) 序列标记层：使用 CRF 层对序列进行标记，预测每个词的实体标签。

(4) 模型训练：使用交叉熵损失和 CRF 损失训练模型。

(5) 预测和后处理：使用训练好的模型进行预测，并使用后处理技术改进预测结果。

该示例演示了如何利用 TensorFlow 实现一个基本的分层 Transformer NER 模型。可以对模型进行进一步改进，如调整超参数、使用更大的预训练模型等，以提高模型性能。

3.3.5　实验分析

HiTRANS 模型描述如图 3.2～图 3.4 所示。

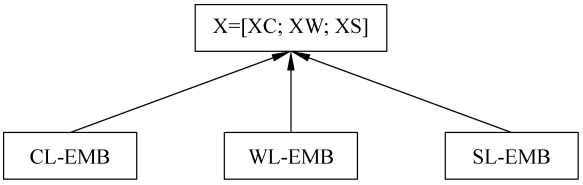

图 3.2 HiTRANS 模型概述——语句的多个层级表征图

为了更好地捕捉句子的语义信息，从多个层级进行表征，例如，字符级、单词级和句子级。字符级嵌入（CL-EMB）、单词级嵌入（WL-EMB）和字符级嵌入（SL-EMB）被级联以获得更好的表示。

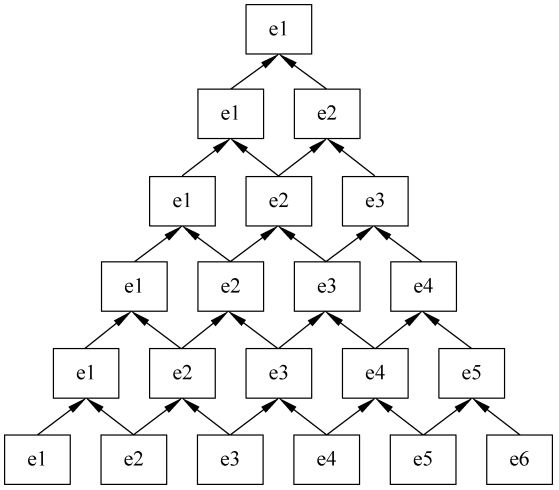

图 3.3 HiTRANS 模型概述——自下而上分层跨度生成模型图

为了从嵌套结构句子中提取嵌套实体，设计了一个分层跨度生成模型，该模型由两个阶段组成，以生成 NER 的候选跨度。具体地，这两个阶段分别以自下而上和自上而下的方式生成候选跨度。在该模型的每一层中，首先利用卷积神经网络（CNN）聚合下一层的两个相邻跨度，生成所有可能的平面实体作为进一步预测的候选。然后利用多头注意力层来增强每个候选的表示学习。自下而上分层跨度生成模型网络的核心思想是通过从底层到顶层递归堆叠卷积神经网络来生成候选跨度的特征向量。

在相反的方向上，由于高层的长实体与低层的短实体在相同的上下文中密切相关，因此高级特征可以通过提供与低级特征互补的附加背景信息来有助于识别低层中的实体。因此，自上而下分层跨度生成模型网络旨在以自上而下的方式将高层信息传播到较低层。P 表示采用 CNN 时的填充。

总体而言，基于超图的方法获得的不错的结果取决于表达性标注模式，然而，模糊性和高时间复杂性几乎不是不可避免的。基于跨度的方法提高了 NER 的性能；然而，它们可能会打破上下文的连续结构。为了缓解这个问题，基于层次的模型通过分

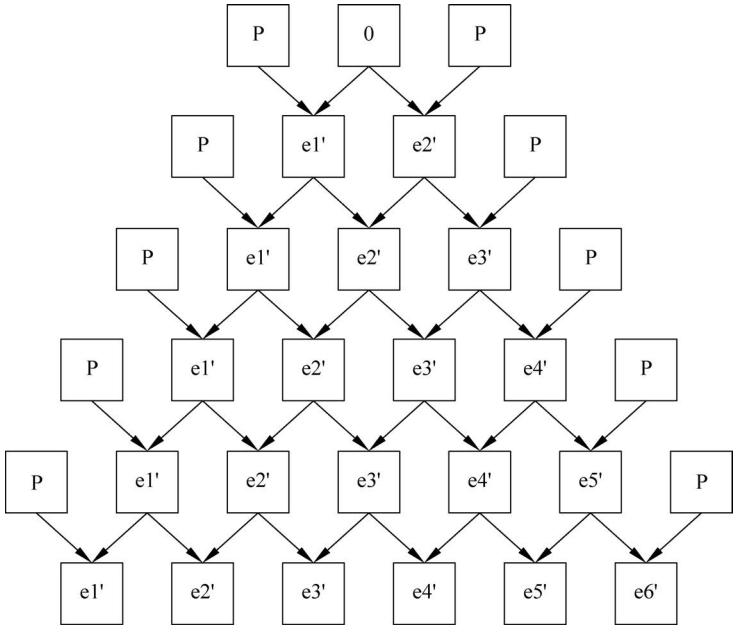

图 3.4　HiTRANS 模型概述——自上而下分层跨度生成模型图

层结构进一步提高了最终的性能，但是，跨度表示过于简化。此外，与预训练语言模型(Pre-trained Language Models，PLMs)结合的方法，例如，BERT 和 ALBERT，通常优于以前的方法，其中利用捕获句子级特征的方法分析上下文结构。

以“中国少数民族古籍总目提要”数据集中回族卷中的识别为例，原文如下：“重修大殿棬棚水房水池碑记石碑 1 通。民国 3 年(1914)买兰馥撰文撰，刘凤舞刻。清真寺重修大殿碑。记述重修大殿棬棚、水房、水池事，落成立碑留念，刻捐资者姓名。碑在今河南省郏县北大街清真寺，总碑面 155cm×60cm，一面有字。刻面 120cm×45cm，竖刻汉文 21 行。碑额 100cm×30cm，刻阿拉伯文 1 行。石材。保存完好。(杨俊杰)”对于这段话可以提取出一些标签信息，如古籍名称、时间和地名等。这段节选的段落的标签抽取结果如表 3.1 所示。

表 3.1　HiTRANS 模型的标签抽取结果

开　始	结　束	文　　本	标　　签
1	12	重修大殿棬棚水房水池碑记	古籍名称
13	14	石碑	类别
15	16	1 通	数量
18	27	民国 3 年(1914)	时间
28	30	买兰馥	编著者
82	92	河南省郏县北大街清真寺	地名
94	106	总碑面 155cm×60cm	大小
113	124	刻面 120cm×45cm	大小

续表

开　始	结　束	文　　本	标　　签
130	132	21 行	大小
134	145	碑额 100cm×30cm	大小
170	173	杨俊杰	编著者

3.4　基于 BERT-CRF 的 NER

3.4.1　引言

随着 NLP 领域的不断发展，利用神经网络进行语言表示的最新进展使得将训练模型的学习内部状态转移到下游任务成为可能，例如 NER 和问题回答。研究表明，利用预训练的语言模型可以提高许多任务的整体性能，并且在标记数据稀缺时非常有益。本节探索了基于特征和微调的 BERT 模型训练策略，微调方法在“中国少数民族古籍总目提要”数据集上获得了新的最先进的结果，在选择性场景（5 个 NE 类）上将 $F1$ 评分提高了 1 分，在总场景（10 个 NE 类）上将 $F1$ 评分提高了 4 分。这些结果表明，BERT-CRF 模型具有很高的性能和迁移能力，可以在 NER 等 NLP 任务中取得优异的表现。

3.4.2　问题引入

NER 的任务是识别提到命名实体的文本范围，并将它们分类到预定义的类别中，例如人员、组织、位置或任何其他感兴趣的类别。尽管概念上很简单，但 NER 并不是一项容易的任务。命名实体的类别高度依赖于文本语义及其周围的上下文。此外，命名实体和评估标准的定义很多，导致了评估复杂性。目前最先进的 NER 系统采用的神经架构已经在语言建模任务上进行了预训练。这类模型的例子有 ELMo、OpenAI GPT、BERT、XLNet、RoBERTa、Albert 和 T5。研究表明，语言建模预训练显著提高了许多 NLP 任务的性能，也减少了监督学习所需的标记数据量。将这些最新技术应用于中文中非常有价值，因为带标记的资源很少，而未标记的文本数据非常丰富。研究人员采用了 BERT（基于变换器的双向编码表示）模型对中文的 NER 任务进行了评估，并比较了基于特征和基于微调的训练策略。这是第一次将 BERT 模型应用于中文的 NER 任务。

3.4.3　相关工作

NER 系统可以基于手工规则也可以基于机器学习方法。对于中文来说，目前的

研究探索了机器学习技术，研究了神经网络模型在中文 NER 中的应用。Vieira 使用从中心词和周围词中提取的 15 个特征创建了一个 CRF 模型。Pirovani 等将 CRF 模型与 Local Grammars 结合起来，采用了类似的方法。从 Collobert 的工作开始，神经网络 NER 系统已经变得流行，因为最小的特征工程要求，这有助于更高的领域独立性。CharWNN 模型通过使用卷积层从每个单词中提取字符级特征，扩展了 Collobert 的工作。这些特征与预训练的词嵌入相连接，然后用于执行顺序分类。LSTM-CRF 架构已被广泛用于 NER 任务。该模型由两个双向 LSTM 组成，用于提取和组合字符级和词级特征，然后由 CRF 层执行顺序分类。

最近的工作探索了与 LSTM-CRF 体系结构一起从语言模型中提取的上下文嵌入。Santos 等使用 Flair Embeddings 从中文语料库上训练的双向字符级 LM 中提取上下文词嵌入。这些嵌入与预训练的词嵌入相连接，并馈送到 Bi-LSTM-CRF 模型。Castro 使用 ELMo 嵌入，该嵌入是 CNN 提取的字符级特征与由 Bi-LSTM 模型组成的双向 LM（biLM）每层隐藏状态的组合。

3.4.4　模型结构

模型体系结构主要由 BERT、CRF 组成，BERT 模型顶部有一个令牌级分类器，然后是一个线性链 CRF。对于 n 个标记的输入序列，BERT 输出一个隐藏维度为 h 的编码标记序列。模型将每个标记的编码表示投影到标签空间，即 $\mathbf{R}^H \rightarrow \mathbf{R}^K$，其中 K 是标签的数量，取决于类的数量和标记方案。然后将分类模型的输出分数 $\boldsymbol{P} \in \mathbf{R}^{n \times K}$ 馈送到 CRF 层，其参数为标签转换矩阵 $\boldsymbol{a} \in \mathbf{R}^{K+2 \times K+2}$。在矩阵 $\boldsymbol{A}$ 中，$\boldsymbol{A}_{i,j}$ 表示从标签 i 到标签 j 的转移得分。$\boldsymbol{A}$ 包含了两个附加状态：序列的开始和结束。如 Lample 所述，对于输入序列 $X=(x_1,x_2,\cdots,x_n)$ 和标签预测序列 $y=(y_1,y_2,\cdots,y_n)$，$y_i \in \{1,2,\cdots,K\}$，则序列的分数定义为

$$s(\boldsymbol{X},\boldsymbol{y})=\sum_{i=0}^{n}\boldsymbol{A}_{y_i,y_{i+1}}+\sum_{i=1}^{n}\boldsymbol{P}_{i,y_i} \tag{3.1}$$

式(3.1)中，y_0 和 y_{n+1} 是开始和结束标记。对模型进行训练，使正确标签序列的对数概率最大化为

$$\log(p(\boldsymbol{y} \mid \boldsymbol{X}))=s(\boldsymbol{X},\boldsymbol{y})-\log\Big(\sum_{\tilde{y} \in Y_X} \mathrm{e}^{s(\boldsymbol{X},\tilde{y})}\Big) \tag{3.2}$$

其中，Y，X 是所有可能的标签序列。式(3.2)中的求和是用动态规划计算的。在求值过程中，通过维特比解码得到最可能的序列。继 Devlin 之后，本节仅计算每个令牌的第一个子令牌的预测和损失。

本节实验了两种迁移学习方法：基于特征和微调。对于基于特征的方法，BERT 模型权值保持不变，只训练分类器模型和 CRF 层。分类器模型由一个 1 层的

Bi-LSTM 和一个线性层组成。没有只使用 BERT 的最后一个隐藏表示层，而是根据 Devlin 对最后 4 层求和。得到的架构类似于 Lample 的 LSTM-CRF 模型，但使用了 BERT 嵌入。对于微调方法，分类器是一个线性层，所有权重，包括 BERT 的权重，在训练期间共同更新。对于这两种方法，没有 CRF 层的模型也被评估。在这种情况下，它们通过最小化交叉熵损失来优化。

为了在计算 BERT 的令牌表示时利用较长的上下文，本节使用文档上下文而不是句子上下文作为输入示例。遵循 Devlin 对 SQuAD 数据集的方法，使用 D 个令牌的跨步将超过 S 个令牌的示例分解为最长为 S 的跨度。在训练过程中，每个跨度被用作一个单独的例子。然而，在求值期间，单个标记 T_i 可以出现在 $N=\frac{S}{D}$ 的多个跨度 s_j 中，因此可能有多达 N 个不同的标签预测 $y_{i,j}$。每个标记的最终预测是从标记更接近中心位置的范围中获取的，即它具有最多上下文信息的范围。图 3.5 说明了评估过程。

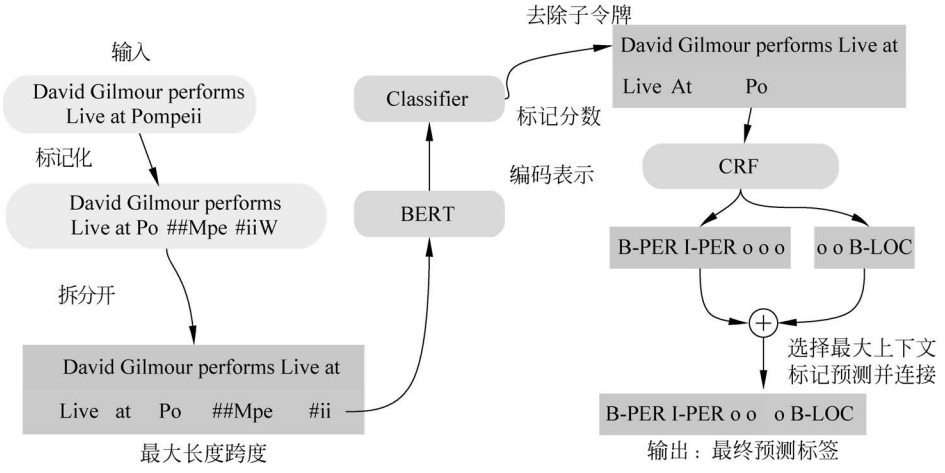

图 3.5 BERT 模型评估过程(见彩插)

3.4.5 实验结果

本节实验中，最大句子长度设置为 $S=512$ 个令牌。本节只训练大小写敏感的模型，因为大写与 NER 相关。使用 sentencepece 和 BPE 算法生成了一个包含 3 万个子词单位的中文词汇表和 20 万个随机中文维基百科文章，然后将其转换为 WordPiece 格式。对于预训练数据，本节使用了 brWaC 语料库，其中包含来自 353 万个文档的 26.8 亿个令牌，是迄今为止最大的开放中文语料库。brWaC 是由完整的文档组成的，它的方法保证了高度的领域多样性和内容质量，这是 BERT 预训练所需要的特征。本节只使用了文档主体，并对数据应用单个后处理步骤，使用 ftfy 库删除 mojibakes3

和残余的 HTML 标签。最终处理的语料库有 17.5GB 的原始文本。预训练输入序列使用默认参数生成，并使用整个工作屏蔽（如果由多个子词单元组成的单词被屏蔽，则其所有子词单元都被屏蔽，并且必须在屏蔽语言建模任务中进行预测）。这些模型被训练了 100 万步。本书使用的学习率是 0.0001，在前 1 万步之后，学习率呈线性衰减。

对于 BERT Base 模型，使用多语言 BERT Base 的检查点初始化权重。在整个训练过程中，本节使用了 128 个批处理大小和 512 个令牌序列。在 TPUv3-8 实例上，该训练需要 4 天，并在训练数据上执行大约 8 次迭代。对于 BERT Large，用英语 BERT Large 的检查点初始化权值。由于它是一个更大的模型，训练时间更长，本节在前 90 万步中使用 128 个令牌的序列，批处理大小为 256，然后在最后 10 万步中使用 512 个令牌的序列，批处理大小为 128。在 TPUv3-8 实例上，该训练需要 7 天，并在训练数据上执行大约 6 次 epoch。用于训练和评估中文 NER 的流行数据集是“中国少数民族古籍总目提要”。表 3.2 包含了数据集一些统计信息。

表 3.2　识别示例（节选自“中国少数民族古籍总目提要”数据集）

数　据　集	文　档　数	符　　号	实　　体
中国少数民族古籍总目提要	45	96593	4635/5436

考虑到文本中的模糊性和不确定性，如句子中的歧义，研究人员对“中国少数民族古籍总目提要”数据集进行了注释。这样，一些文本段包含< ALT >标记，这些标记包含多个可选的命名实体标识解决方案。此外，可以将多个类别分配给单个命名实体。为了将 NER 建模为序列标记问题，须为每个未确定的段或实体选择一个单一的真理。要解析数据集中的每个< ALT >标记，本节的方法是选择包含最多命名实体的替代方案。在命名实体存在同样多的情况下，选择第一个。要解析每个分配了多个类的命名实体，只需为场景选择第一个有效的类。数据集预处理脚本在 GitHub4 上提供。

比较了模型在两种情况下（完全和选择性）的性能。所有指标都是使用 CoNLL 2003 评估脚本计算的，该脚本由实体级微 $F1$ 评分组成，只考虑精确匹配。本节讨论的 BERT-CRF 模型优于之前的模型（Bi-LSTM-CRF＋FlairBBP），在选择性场景中将 $F1$ 评分提高了约 1 分，在总场景中将 $F1$ 评分提高了 4 分。有趣的是，Flair 嵌入在英语 NER 上优于 BERT 模型。与没有上下文嵌入的 LSTM-CRF 架构相比，本节的模型在总场景和选择场景的 $F1$ 评分上分别高出 8.3 分和 7.0 分。中文 BERT（PT-BERT-BASE 和 PT-BERT-LARGE）也优于以前的结果，即使没有强制执行 CRF 层提供的顺序分类。在比较总体 $F1$ 评分时，具有 CRF 的模型与其更简单的变体改进或执行相似。在大多数情况下，它们显示出更高的精确率、更低的召回率。

虽然中文 BERT large 模型在这两种情况下都是表现很好的，但当在基于特征的

方法中使用时，它们的性能会下降，比它们的较小变体表现得更差，但仍然比多语言BERT好。此外，可以看出，与BERT base模型相比，BERT large模型并没有给选择场景带来太大的改善。假设这是由于NER数据集的规模较小。与微调方法相比，基于特征的方法的模型表现明显较差。研究发现，表现远高于英语语言的NER报告值。对于IOB2方案，滤除无效过渡的后处理步骤平均使基于特征的方法和微调方法的$F1$评分分别提高1.9分和1.2分。这一步骤使召回率降低了0.4%，但平均而言，精确率提高了3.5%。

例如在蒙古族卷文书中的一段："鄂尔多斯右翼中旗札萨克贝勒索诺木喇布斋根敦札文1件。2页。清嘉庆四年(1799)索诺木喇布斋根敦撰。蒙古文。记登记鄂尔多斯后翼中旗(今鄂托克旗)所有寺庙造册事宜。鄂尔多斯右翼中旗札萨克贝勒索诺木喇布斋根敦致伊克昭盟鄂尔多斯右翼后旗(今杭锦旗)札萨克贝子喇什达尔济札文。文中记载申请鄂尔多斯右翼中旗直辖光缘寺及所有已命名的寺庙进行重新登记造册之事。对研究蒙古族佛教文化有史料价值。全宗名《清朝杭锦旗政府》。全宗号57，目录号1，卷号113，件号1，第7～8页。麻纸，毛笔楷体，墨书，页面24.1cm×12.2cm+6折。钤有鄂尔多斯右翼中旗大红印。残缺。今藏鄂尔多斯市档案馆(巴音登录韩长寿、朝伦巴特尔、巴音译)。"对于这段话可以提取出一些标签信息，如古籍名称、时间和收藏单位等。标签抽取结果如表3.3所示。

表3.3 BERT模型的标签抽取结果

开　始	结　束	文　本	标　签
1	23	鄂尔多斯右翼中旗札萨克贝勒索诺木喇布斋根敦札文	古籍名称
31	46	清嘉庆四年(1799)	时间
263	288	鄂尔多斯市档案馆	收藏单位

本节通过在大量未标记文本的语料库上预训练中文BERT模型，并对中文NER任务上的BERT-CRF模型进行微调，在"中国少数民族古籍总目提要"语料库上讨论分析了一种新的技术。尽管它是在更少的数据上进行预训练，但实验结果表明，BERT-CRF模型优于之前模型(Bi-LSTM-CRF+FlairBBP)的效果。

3.5 基于迁移学习的细粒度BERT的NER

3.5.1 引言

在如今的数字化时代，对于古籍文本的处理和研究具有重要的学术和文化价值。然而，由于古籍文本的特殊性和复杂性，传统的文本识别方法往往难以达到理想的效果。近年来，基于深度学习的技术在NLP领域取得了巨大的进展，为古籍文本识别带

来了新的希望。特别是基于 Transformer 架构的 BERT 模型的出现，在处理自然语言数据方面取得了突破性的成果。然而，面对古籍文本的特殊挑战，传统的 BERT 模型仍然存在一些局限性。为了解决这一问题，迁移学习成为古籍文本识别中的一种有效方法。迁移学习通过利用其他领域的知识和数据来提升目标领域任务的性能。在古籍文本识别中，迁移学习的思想被成功地应用于提高模型的识别准确度和泛化能力。通过预训练一个大规模的 BERT 模型，在其他相关领域的大量文本数据上学习丰富的语言知识，然后通过微调的方式，在古籍文本识别任务上进行训练，可以有效地利用跨领域的知识，提高模型对于古籍文本的识别能力。

3.5.2　问题引入

在中文 NER 方面，Yin 提出了一种基于部首级特征和自注意机制的 Bi-LSTM-CRF 来解决中文临床 NER 问题。He 针对中文社交媒体中的 NER 问题提出了一个统一的特征模型，该模型可以从外国语料库和该领域的未注释文本中学习。Yin 提出了一种考虑模糊实体边界的标注策略，结合领域专家知识，构建了基于微博数据的 MilitaryCorpus。在双向编码器词向量表达层和在 BERT 的指导下，利用该模型获得词级字符。在 Bi-LSTM 的指导下，该层提取上下文特征，形成特征矩阵，最后用 CRF 生成最优标签序列。

Shen 将深度学习与主动学习相结合，以减少标记的训练数据的数量。引入了一种轻量级的 NER 方法，即 CNN-CNN-LSTM 模型，以加快操作。该模型由一个 CNN 字符编码器、一个 CNN 单词编码器和一个 LSTM 标签解码器组成。LukaGligic 引导神经网络(NN)模型通过迁移学习对未注释的电子健康记录(EHR)上执行的次要任务进行词嵌入预训练，并将输出嵌入作为一系列 NN 架构的基础。Van CuongTran 组装了一种方法，使用主动学习(AL)和自我学习，通过使用机器标记和手动标记的数据来减少推文流中 NER 任务的工作负荷。CRF 也被选为高度可靠案例的算法。Kung 建立了一个基于普通话 NER 模块的迁移学习系统，在灾害管理中对损失信息进行收集和分析。针对文物区缺乏标签数据的情况，张晓明提出了一个模型——文物 NER 的半监督模型，这个模型利用没有标签数据训练的 Bi-LSTM 和 CRF 模型，获得了有效的识别性能。

特定领域 NER 的主要困难在于缺乏规范的标注语料，然而，深度网络模型的训练通常需要一个大的注释语料库来训练。因此，深度网络模型直接应用于某一特定领域的作用往往不大。本节通过结合主动学习和迁移学习，提出了一种考虑主动学习的模型迁移方法，对 NER 的研究基于“中国少数民族古籍总目提要”数据集，在古籍研究领域中应用 NER 技术是很有必要的，古籍文本往往具有复杂的语言形式和特定的文化背景，而其中蕴含的丰富实体信息对于深入理解历史、文化和社会等方面信息具

有重要意义，应用 NER 技术可以有效地自动化实体识别的过程，通过识别和标注古籍文本中的命名实体，包括人名、地名、机构名、日期等，从而提供更高效、准确的文本分析和信息抽取。研究者可以更加方便地进行进一步的文本分析、信息检索和知识发现。通过预训练和微调 BERT 模型，利用现代文本数据的丰富语言表示，该方法能够提高古籍文本的 NER 准确度和稳健性，为古籍文献的数字化处理和研究提供了有力的工具和技术支持。

3.5.3 实验过程

在本节提出的模型的训练过程中，采用了主动学习和自我学习相结合的方法，并选择 CRF 层的条件概率作为 CRF 层预测的置信度(nx)。在迭代过程中，置信度高于阈值的样本直接作为训练样本加入；置信度低于阈值的样本则交给人工标注。这将解决领域语料库数据的问题，并提高模型的普适性。考虑激活学习方法的 ALBERT-AttBiLSTM-CRF 模型迁移如图 3.6 所示。

算法 考虑激活学习的模型迁移(L、U、M、Conf)

输入：
L：有标记的训练数据集；
U：未标记的训练数据集；
M：拟议模式；
Conf：置信水平。
输出：训练好的模型 M'。
1：//模型转移
2：使用训练集 L 进行训练，得到模型 M
3：使用模型 M 预测未标记的数据集 U
4：计算模型 CRF 层的条件概率的得分($Y|X$)作为置信水平 Conf
5：//主动学习
6：对每个样本的每个置信度执行操作
7：选择所有 Conf$\geqslant$Conf$_{high}$ 的样本 U_{high}，将其加入 L($L=L+U_{high}$)，并从 U($U=U-U_{high}$)中删除 U_{high}
8：选择所有 Conf$<$Conf$_{low}$ 的样本 U_{low}，对其进行手动重新注释，将其加入 L($L=L+U_{low}$)，并从 U($U=U-U_{low}$)中删除 U_{low}
9：按照上述步骤迭代 n 次，直到模型 M'收敛
10：**结束**
11：**返回**训练过的模型 M'

图 3.6 考虑激活学习方法的 ALBERT-AttBiLSTM-CRF 模型迁移

本节采用基于轻量级多网络协作和主动学习的新型中文细粒度 NER 方法进行识别。首先，ALBERT 被用来提取文本数据的词向量，一个 AttBiLSTM 被用来从输入向量中捕捉特征，并获得文本的上下文信息。前一个网络输出的特征矩阵由 CRF 标记，以获得标记的序列。然后，提出了一种考虑主动学习的模型转移方法，通过在源

域上训练得到的模型转移到目标域，得到一个新的模型。在新模型的基础上，主动学习被用来标记未标记的数据，以不断增加数据量并提高模型的质量。

为了验证所提出的方法的有效性，本节设计了相关的比较实验，将其与主流的NER方法进行比较。此外，K-fold交叉验证法被用来将语料库数据集划分为10个相等的部分，每个部分都通过分层抽样得到。实验通过轮换9部分进行训练，剩余数据进行测试，评估的指标是召回率(R)、精确率(P)和$F1$评分，如式(3.3)～式(3.5)所示。最后，将10个实验的结果相加，再取平均值，可以作为模型优化的一个指标：

$$\text{召回率}=\frac{\text{TP}}{(\text{TP}+\text{FN})}\times 100\% \tag{3.3}$$

$$\text{精确率}=\frac{\text{TP}}{(\text{TP}+\text{FP})}\times 100\% \tag{3.4}$$

$$F1\text{ 评分}=\frac{2\times \text{精确率}\times \text{召回率}}{\text{精确率}+\text{召回率}}\times 100\% \tag{3.5}$$

本书的“中国少数民族古籍总目提要”数据集包含45本已经标注好的估计文本书件，其中包括古代的书籍、文献和手稿，通常包含了各种领域的知识、历史事件、文学作品等，本书对其中古籍故事中人物、地点、作者、收藏地等十多个信息进行了标注，用于实体识别研究。这个数据集的特点在于它聚焦于中国的少数民族古籍文献，为研究者提供了深入了解和探索中国多元文化遗产的机会。通过对古籍的研究，人们可以更好地理解古代社会和思想，挖掘宝贵的文化遗产，以及推动学术研究和文化传承。

获得相应的文本数据后，本节对语料库进行注释。注释使用YEDDA(轻量级协作文本跨度注释工具)系统对8种类型的命名实体进行手工注释，语料库为中文。为了获得高质量的标签预测结果并添加相应的约束条件，本节使用BIO注释方法。本节将每个元素注释为“B-实体”、“I-实体”或“O”。“B-实体”表示该片段是X类型的，并且该项目位于粒子的开头。“I-实体”意味着该片段是X类型的，并且该元素位于片段的中间。“O”表示该片段不属于任何类型。

3.5.4　实验结果

ALBERT-AttBiLSTM-CRF与其他基准算法一起应用于“中国少数民族古籍总目提要”数据集，NER的结果如表3.4所示。可以看出，ALBERT-AttBiLSTM-CRF在P、R和$F1$评分方面优于CLUENER2020细粒度数据集发布者提供的RoBERTa的最佳结果。特别是它的$F1$评分高出5.4%。为了彻底验证不同ALBERT版本对实体识别效果的影响，本节使用了微小、基本、大和xlarge四个版本进行比较。表3.4显示了不同算法在“中国少数民族古籍总目提要”数据集上的得分情况。可以看出，整体效果并没有随着模型预训练语料库大小和参数的增加而越来越好。最好的结果出现在大版本中，其中精确率、召回率和$F1$评分分别为0.9253、0.8702和0.8962。这

些结果比提议的基准的最佳结果分别高 13.27%、5.33%和 9.2%。

表 3.4 基于“中国少数民族古籍总目提要”数据集上不同算法的比较

方 法	精 确 率	召 回 率	*F*1 评分
Bi-LSTM-CRF	0.7106	0.6120	0.6936
ALBERT-Bi-LSTM-CRF	0.8876	0.8270	0.8555
ALBERT-CRF	0.8094	0.6897	0.7000
ALBERT-Bi-LSTM	0.7736	0.8132	0.7925
En2Bi-LSTM-CRF	0.9156	0.8337	0.8720
ALBERT	0.7992	0.6459	0.7107
BERT	0.7724	0.8046	0.7882
RoBERTa	0.7926	0.8169	0.8042
ALBERT-AttBiLSTM-CRF (our)	0.9253	0.8702	0.8962

从前面的实验结果可知，ALBERT-AttBilSTM-CRF(our)在“中国少数民族古籍总目提要”数据集上取得了最好的结果，所以在模型转移部分采用了 ALBERT-AttBiLSTM-CRF(our)模型。

本节使用了一个精简的预训练模型对文本数据进行阶段性的文本嵌入表示；使用 Bi-LSTM 对文本输入进行特征提取，这样可以有效地捕捉文本的上下文信息，提取的特征输入被随机分配到现场网络，以获得相应实体的文本数据。随后，本节通过结合主动学习和迁移学习，提出了一种考虑主动学习的模型迁移方法。该方法利用公共数据集对提出的 ALBERT-AttBiLSTM-CRF(our)模型进行预训练，并对领域数据进行标记以调整模型参数，对未标记的领域数据进行主动学习，用于优化名为实体识别的领域的结果。为了验证所提出 ALBERT-AttBiLSTM-CRF(our)方法的有效性，该方法与 BERT、RoBERTa 等主流方法在“中国少数民族古籍总目提要”数据集上进行了验证；与目前最佳基准 RoBERTa-wwm-large-ext 相比，结果的准确率提高了 9.2%。使用在“中国少数民族古籍总目提要”数据集上取得最佳结果的 ALBERT-AttBiLSTM-CRF(our)模型作为源模型，本节使用 MTAL 迁移到 Manufacturing-NER 数据集。迁移结果显示改进了 3.55%，证明了 ALBERT-AttBiLSTM-CRF 模型迁移考虑激活学习方法的有效性。

例如在蒙古族卷文书中的一段：“阿拉善旗札萨克多罗贝勒罗卜藏多尔济咨文 1 件。1 页。清乾隆二十一年(1756)三月罗卜藏多尔济撰。蒙古文。阿拉善旗札萨克多罗贝勒罗卜藏多尔济致全旗台吉、塔布囊、喇嘛咨文。文中记载准许甘珠尔巴喇嘛在阿拉善旗境内募化之事。对研究蒙古族佛教文化有史料价值。全宗名《清代阿拉善旗档案》，全宗号 101，目录号 3，卷号 58，件号 23，第 68 页。纸，楷体，墨书，页面 31cm×28cm。保存完好。今藏阿拉善左旗档案馆。(乌如希拉特登录赛音朝格图译)”对于这段话的抽取结果如表 3.5 所示。

表 3.5　BERT 模型标签抽取结果

开　始	结　束	文　本	标　签
1	19	阿拉善旗札萨克多罗贝勒罗卜藏多尔济咨文	古籍名称
26	40	清乾隆二十一年(1756)三月	时间
194	201	阿拉善左旗档案馆	收藏单位