

分 类 号：TP391.1
研究生学号：2201903026

单位代码：10190
密 级：公 开

長春工業大學
碩 士 学 位 论 文

郑肇谦

2022 年 6 月



基于深度学习的中文实体关系 抽取研究

Deep Learning Based Chinese Entity Relation Extraction Research

硕 士 研 究 生：郑肇谦

导 师：赵 辉教授

申 请 学 位：工学硕士

学 科：计算机科学与技术

所 在 单 位：计算机科学与工程学院

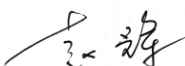
答 辩 日 期：2022 年 6 月

授予学位单位：长春工业大学

长春工业大学硕士学位论文原创性声明

本人郑重声明：所呈交的硕士学位论文《基于深度学习的中文实体关系抽取研究》，是本人在指导教师的指导下，独立进行研究工作所取得的成果。除文中已经注明引用的内容外，本论文不包含任何其他个人或集体已经发表或撰写过的作品成果，不存在学位论文买卖、代写、抄袭等学术不端行为。对本文的研究做出重要贡献的个人和集体，均已在文中以明确方式标明。本人完全意识到本声明的法律结果由本人承担。

作者 签名： 鄭肇謙

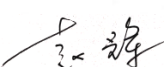
指导教师签名： 

日 期： 2022 年 6 月 2 日

长春工业大学硕士学位论文版权使用授权书

本学位论文作者及指导教师完全了解“长春工业大学研究生学位论文版权使用规定”，同意长春工业大学保留并向国家有关部门或机构送交学位论文的复印件和电子版，允许论文被查阅和借阅。本人授权长春工业大学可以将本学位论文的全部或部分内容编入有关数据库进行检索，也可采用影印、缩印或扫描等复制手段保存和汇编学位论文。保密的论文在解密后遵守此规定。

作者 签名： 鄭肇謙

指导教师签名： 

日 期： 2022 年 6 月 2 日

摘 要

随着互联网技术不断发展和海量数据不断涌现,如何从海量非结构化数据中提取有用的结构化信息成为现阶段的研究热点,关系抽取应运而生。关系抽取作为上游提供基础数据的技术,在下游诸多领域都具有重要的应用价值,如知识图谱、语义理解、推荐检索、机器翻译和智能问答等。近几年,深度学习模型已经成为关系抽取的最先进方法,现有的工作也取得了相当大的成果,但还存在实体嵌套、关系重叠、暴露偏差等问题,严重影响了关系抽取模型的精度,本文从解决上述问题的角度出发,提出了两种实体关系联合抽取模型,主要工作内容如下几个方面:

1. 针对现有实体关系抽取方法中存在的实体嵌套问题,区别于原有基于词元(Token)进行关系抽取的思路,采用基于片段(Span)的思路进行关系抽取,并且设计和使用滑动窗口和三种映射策略将词元序列进行组合排列重新平铺成片段序列。

2. 针对现有实体关系抽取方法中存在的暴露偏差和关系重叠等问题,提出了一种基于片段多头选择的实体关系联合抽取方法(Span based Multi Head Selection, SMHS),将实体关系抽取转化为片段级的多头关系选择问题。首先通过片段标记器、片段嵌入的方式构造片段语义向量,结合所提出的片段映射策略将原本的词元序列转化为片段序列,然后利用 LSTM、多头自注意力机制进行片段特征提取,最后使用多头选择机制进行片段级关系解码且引入片段分类任务辅助训练,单步解码出关系三元组。

3. 针对 SMHS 时间复杂度较大,推理速度较慢,提出了一种基于片段标注的实体关系联合抽取模型(Span-Labeling Based Model, SLM),将实体关系抽取问题转化为片段标注问题。首先同样地通过将词元向量转化为片段语义向量,结合片段映射策略将词元序列转化为片段序列,然后利用 GRU、多头自注意力机制进行片段特征抽取,最后利用精心设计的片段关系标签进行关系标签分类,单步解码出关系三元组。

4. 基于权威中文关系抽取数据集 DuIE2.0 进行实验,且重新对数据集的标注形式进行修改。为验证模型性能,选取了当前主流的关系抽取模型进行对比实验;为验证所提出模块的有效性,进行消融实验;为探究模型参数对模型的影响,进行影响因素实验。实验表明,本文所提出的两个模型取得了比当前主流抽取方法更好的效果;所提出模块对模型性能确有提升作用;确定了相关参数对模型的潜在影响,验证了模型的有效性和优越性。两个模型比较而言,SMHS 的精度较 SLM 高,但 SLM 在时间空间复杂度和推理速度方面占据优势。

关键词: 关系抽取 联合抽取 片段抽取 暴露偏差 关系重叠

Abstract

With the continuous development of Internet technology and the continuous emergence of massive data, how to extract usefully structured information from massive unstructured data has become a research hotspot at this stage, and relation extraction has emerged as the times require. As an upstream technology for providing basic data, relation extraction has important application value in many downstream fields, such as knowledge graph, semantic understanding, recommendation retrieval, machine translation, and intelligent question answering. In recent years, deep learning models have become the state-of-the-art method for relation extraction, and existing work has achieved considerable results, but there are still problems such as entity nesting, relation overlap, and exposure bias, which seriously affect the performance of relation extraction models. From the perspective of solving the above problems, this paper proposes two entity-relationship joint extraction models. The main work includes the following aspects.

1. Aiming at the entity nesting problem existing in the existing entity relation extraction methods, different from the original idea of relation extraction based on token, this paper adopts the idea of span for relation extraction, and designs and uses it. Sliding windows and three mapping strategies recombine and rearrange the token sequences into span sequences.

2. Aiming at the problems of exposure bias and relationship overlap existing in existing entity relationship extraction methods, a joint entity relationship extraction method (Span based Multi Head Selection , SMHS) based on span multi-head selection is proposed, which converts entity relationship extraction into span-level multi-head relationship selection question. First, the span semantic vector is constructed by means of span marker and span embedding, and the original token sequence is converted into span sequence by combining the proposed span mapping strategy, and then LSTM and multi-head self-attention mechanism are used to extract span features. The selection mechanism performs segment-level relation decoding and introduces span classification task to assist training, and decodes relation triples in a single step.

3. Aiming at the large time complexity and slow reasoning speed of SMHS, an entity relation extraction model (Span-Labeling Based Model , SLM) based on span-labeling is proposed, which transforms the entity relation extraction problem into the span-labeling

problem. First, convert the token vector into span semantic vector, and combine the span mapping strategy to convert the original token sequence into span sequence, then use GRU and multi-head self-attention mechanism to extract span features, and finally use carefully designed span relationship. The labels are classified by relation labels, and the relation triples are decoded in a single step.

4. Based on the experiments on the authoritative Chinese relation extraction dataset DuIE2.0, the labeling form of the dataset is re-modified. To verify the performance of the model, the current mainstream relation extraction model was selected for comparative experiments; in order to verify the effectiveness of the proposed module, an ablation experiment was carried out; in order to explore the influence of model parameters on the model, an experiment of influencing factors was carried out. Experiments show that the two models proposed in this paper have achieved better results than the current mainstream extraction methods; the proposed modules can indeed improve the performance of the model; the potential impact of related parameters on the model is determined, and the effectiveness and efficiency of the model are verified. superiority. Comparing the two models, the accuracy of SMHS is higher than that of SLM, but SLM has advantages in terms of time and space complexity and reasoning speed.

Key words: Relation extraction Joint extraction Span extraction Exposure bias
Relation overlap

目 录

摘 要.....	I
ABSTRACT.....	II
第 1 章 绪 论.....	1
1.1 课题研究背景和意义.....	1
1.2 关系抽取研究现状.....	2
1.2.1 传统方法.....	2
1.2.2 深度学习方法.....	5
1.2.3 存在问题.....	6
1.3 本文主要工作.....	7
1.4 本文章节架构.....	8
第 2 章 相关技术基础.....	10
2.1 预训练模型.....	10
2.1.1 静态词嵌入预训练模型.....	10
2.1.2 动态词嵌入预训练模型.....	11
2.2 神经网络模型.....	13
2.2.1 循环神经网络 (RNN).....	13
2.2.2 长短时记忆网络 (LSTM).....	14
2.2.3 门控循环单元 (GRU).....	16
2.3 多头自注意力机制.....	17
2.4 片段标记和片段映射策略.....	19
2.5 本章小结.....	20
第 3 章 基于片段多头选择的实体关系联合抽取模型.....	22
3.1 模型概述.....	22
3.2 编码层.....	23
3.2.1 词元编码层.....	23
3.2.2 片段编码层.....	23
3.2.3 LSTM 和多头自注意力机制层.....	24
3.3 解码层.....	25
3.3.1 多头选择机制层.....	25

3.3.2 片段分类层.....	26
3.4 损失函数.....	27
3.5 实验与分析.....	27
3.5.1 数据集.....	27
3.5.2 实验环境与参数设置.....	30
3.5.3 实验评价标准.....	30
3.5.4 实验结果分析.....	31
3.6 本章小结.....	35
第4章 基于片段标注的实体关系联合抽取模型	36
4.1 模型概述.....	36
4.2 标签设计.....	37
4.3 编码层.....	37
4.3.1 词元编码层.....	38
4.3.2 片段编码层.....	38
4.3.3 GRU 和多头自注意力机制层	39
4.4 解码层.....	40
4.5 损失函数.....	40
4.6 实验与分析.....	40
4.6.1 数据集介绍.....	40
4.6.2 实验环境与参数设置.....	41
4.6.3 实验评价标准.....	41
4.6.4 实验结果分析.....	41
4.7 本章小结.....	45
第5章 总结与展望.....	46
5.1 总结与结论.....	46
5.2 展望.....	47
参考文献.....	48
致 谢.....	53
作者简介.....	54
攻读硕士学位期间研究成果	55

第1章 绪论

1.1 课题研究背景和意义

随着互联网技术快速发展和互联网用户人数激增，导致海量的数据涌入互联网，由于大部分数据并不是结构化数据，例如复杂文本、图像、音视、视频等。由于这些数据本身分散，冗余异构的原因，导致传统的数据挖掘工具无法快速有效地挖掘出有效信息。面对内容丰富的数据却无法从中获取有价值的信息，因此如何从大量数据中准确、高效地找到所需内容，并解析这些内容，转化为可利用的结构化数据，成为了文本挖掘领域迫切需要解决的关键问题，信息抽取便由此诞生^[1]。

信息抽取旨在对文本信息进行深层挖掘，由于自然语言没有固定的结构规律可循，所以计算机处理和理解文本信息较为困难，研究人员不断攻坚克难，信息抽取的研究得以快速发展。实体关系抽取作为信息抽取的重要子任务之一，受到诸多研究人员的关注，其目的是识别出文本中所包含的实体信息（如书名、人名、国家名、机构名等）和实体之间存在的关系，构建形如（主实体，关系，客实体）的实体关系三元组，由这些实体关系三元组相互链接进而组成的大型知识图谱网络在其他领域中得到了广泛应用^[2]，在问答系统、智能对话、智能搜索，等诸多下游任务有较强的应用价值。

关系抽取具有重要的应用价值，被应用在诸多领域中，包括语义理解、词义消歧、智能搜索、机器翻译、智能问答等，关系抽取为其提供基础数据。随着知识图谱的深入研究和广泛应用，关系抽取作为对知识图谱构建质量具有决定性作用的核心技术也是备受关注，吸引了许多研究人员参与，各类具有不同范式的新方法和新模型不断涌现，关系抽取技术的性能也不断提升^[3]。

由于关系抽取的应用价值前景良好，本文旨在运用深度学习的成熟技术，结合目前自然语言处理领域非常火热的预训练模型（谷歌的 BERT，百度的 ERNIE），提出端到端的开放式的联合关系抽取模型。本文提出的模型采用业界领先的预训练模型来构成 Embedding 层，结合深度学习中的较为常见的用于提取语义信息的多头自注意力机制、LSTM、GRU 等结构，采用基于片段（Span）级别的解码，能够单步解码出实体关系三元组。模型在使用时，无需对文本进行预处理，方便使用，准确率高，且根据实际情况适当调整超参数，平衡精度和推理、训练速度，应用范围广泛，可迁移性强，可以在大量的非结构化的文本中，快速抽取出实体关系三元组，快速构成一个小型知识图谱，进而为其相关应用提供数据基础。因此本课题具有较强的应用价值和现实意义。

1.2 关系抽取研究现状

关系抽取研究的历史并不久远，最早在 1998 年的第七届信息理解会议（message understanding conference, MUC）上，首次提出模板抽取任务，并且设计了相应的评价体系，后发展为关系抽取^[2,3]。在 1999 年，美国国家技术研究院(National Institute of Standards and Technology, NIST)召开的自动内容抽取会议(automatic content extraction, ACE)替代了 MUC 会议，并对关系抽取任务的评测标准进行细化和训练数据进行扩充。而后在 2002 年的第三届会议加入了实体关系发现和识别任务(relation detection and recognition, RDR)。2008 年，ACE 会议将关系抽取任务中的关系类型划分为 7 种。2009 年，ACE 会议纳入文本分析会议（text analysis conference, TAC），关系抽取加入了知识库总体（knowledge base population, KBP）任务，成为其重要部分。后期出现的语义评估(semantic evaluation, SemEval)会议 SemEval-2007 中，在评测任务四中设定了七种常见实体间的实体关系，而后在 SemEval-2010 评测任务八中将类型扩充到了十种，进而掀起了实体关系抽取研究的新热潮^[1,4]。

国内的相关研究起步稍晚，主要是中国科学院大学、清华大学、哈尔滨工业大学等诸多科研院校在推动其发展上做出重要工作。中文与英文等语言在结构和语义、语法、句法方面差别显著，这导致中文实体关系抽取效果不如英文等其他语言。而且目前中文语料库稀少，并且创建中文语料库的难度较大，需要中文分词、词性标注和句法、语法分析等预处理过程，在此过程中存在不可避免的误差和错误，费时费力。

中文领域的实体关系抽取研究具有较大的挑战性，主要有：中文存在词汇边界模糊的问题，易造成边界歧义和边界错误，大大加剧了关系抽取的难度；中文句式表达灵活复杂、多省略，不同领域中汉语重载、一词多义（如高富帅、老古董等），代词指代不明确，实体嵌套等问题，甚至出现文言文，这都使得中文实体关系抽取，更为困难^[5]。

目前，实体关系抽取方法根据所用技术的不同，主要分为两个大类，一是传统方法，基于模板和词典的规则匹配方法和基于人工构造文本特征工程的传统机器学习方法；二是深度学习方法，基于各类设计好的神经网络自动提取文本特征。由于深度学习方法的精度高，速度快，端到端无需处理文本等优点，以及计算机硬件和神经网络的发展，当前深度学习方法已成为主流的实体关系抽取方法。

1.2.1 传统方法

早期中文实体关系抽取，通过人工构造和语义规则来进行实体关系抽取。基于规则的方法要提前定义若干个基于词法、词义、词性的规则模式集合，利用这些规则模

式集合去描述实体间关系，这需要定义的人员，拥有大量相关的语言学知识。关系抽取时，将经过预处理的文本片段与提前规定好的规则模式进行匹配对应，而后进行实体关系判别，最后完成实体关系的抽取^[1]。中文领域基于规则的关系抽取起步较晚。相较英文，邓肇等^[6]在模式匹配基础上加入对词汇、语义等语言特征元素，进行中文实体关系抽取。温春等^[7]提出了一种扩展的关联规则方法用于抽取中文非分类关系，能够得出具体的非分类关系名称。

基于规则的实体关系抽取方法，在特定领域和特定的语料库取得了一定的成果，但缺点也是显而易见，需要规则模式的构建者对相关领域的知识背景和特点都有深入了解，且还需要对语言学知识有相当的基础，对领域内的关系模式进行穷举，模型缺乏可移植性、语料人工标注成本高、模型召回率较低等。这又使得研究人员尝试词典驱动，基于本体等方式来进行关系抽取。基于词典驱动的实体关系抽取方法使用 KMP 等字符串匹配算法识别实体，并结合词典中的动词和其所对应的语言关系构成，来识别关系类型，最后完成关系抽取，抽取时需不断对词典进行扩充和新增指示动词，进行迭代完善。

基于传统机器学习的关系抽取方法将实体关系抽取任务视为实体关系分类问题。首先需要对目标语料中的所有目标实体和目标关系，进行人工标注，进行筛选，构建训练和测试数据集，然后使用训练数据集训练的出的实体关系分类器对测试集中的候选实体及关系进行识别。主要包括：基于特征向量的方法和基于核函数的方法^[8]。

基于特征向量的方法，首先通过从文本上下文中提取有用信息（词汇特征、句法特征、语义特征等）来构建文本特征向量，将文本特征向量输入到模型中，而后利用各类机器学习中的分类算法来训练实体关系抽取模型。

Kambhatla^[10]结合词汇、句法和语义构造的特征向量，采用最大熵模型来进行关系分类，实验证明了结合多方面的语言特征进行关系抽取是十分有价值的，也为后续类似研究奠定了基础。Zhou 等^[11]借鉴了 Kambhatla 的思路，在其基础之上通过加入 Word Net、基本词组块和 Name List 等手段来增强文本语义特征，并采用 SVM（支持向量机）来进行实体关系抽取。Jiang 等^[12]系统地研究了一个大的特征空间来提取关系，并评估了不同特征子空间的有效性，实验表明，将不同复杂度级别的特征组合起来，再结合面向特定任务的特征剪枝，分类器能获得最佳性能。在中文领域，车万翔等^[13]使用 Winnow 和 SVM 算法进行实体关系抽取实验，实验表明在使用相同的特征集，不同的机器学习分类算法对实体关系抽取的最终性能影响较小，而应当将集中精力寻找好的特征，来帮助实体关系抽取模型提升性能。郭喜跃等^[14]将句法分析和语义分析的结果加入到实体关系的特征之中，增强实体信息表达，实验结果表明此方法效果明显。高俊平等^[15]提出一种面向中文维基百科领域知识的演化关系抽取方法，使

用 CRF（条件随机场）并利用语法分析特征，构建演化关系推理模型。模型结合序列标注，以词性、命名实体类别、依存句法结构以及语义角色等语言特征作为模式属性，能有效识别领域中概念间的演化关系，具有较高精度，更适用于中文领域知识演化的实体关系抽取。甘丽新等^[16]从语义角度提出最近句法依赖动词特征和趋向核心动词特征，句内基于协陪义动词的隐式关系推理规则。设计了协陪义候选句型分类算法以及相应的协陪义成分识别算法。

虽然基于特征向量的实体关系抽取方法取得了不错成果，但存在较大局限性，该方法十分依靠特征工程的工作，特征工程的好坏直接影响模型性能，各类语言、文本特征已被使用殆尽，无法挖掘出新的特征，难以提高模型性能。

采用基于核函数的关系抽取方法，使用隐性特征映射代替显性的特征映射，对高维特征空间的样本只需计算其内积节省了人工提取特征的步骤，拓展了基于构造特征向量的实体关系抽取的技术方法^[17]。

Zelenko 等^[18]首先在实体关系抽取中，将核函数结合浅层解析树结构进行关系抽取，且使用 SVM（支持向量机）和投票感知模型等分类算法。Cullota 等^[19]融合依存树函数和知识库 Word net，同时使用 SVM 分类算法，提出了扩展子树节点间的匹配算法。Zhang 等^[20]首次融合多个单一核函数，把复合核函数运用在实体关系抽取任务中。在中文领域方面，刘克彬等^[21]将改进的语义序列核函数应用于中文实体关系自动抽取系统，并结合 KNN 分类器进行分类并标注关系类型。虞欢欢等^[22]提出了一种基于卷积树核的汉语语义关系推理方法，显著提高了语义关系抽取的性能。陈鹏^[23]提出一种平面凸组合核函数和一种融合领域知识的树核函数，并使用融合平面核的多核融合进行中文领域的实体关系抽取。基于核函数的方法，虽然在一定程度上能省去构建高维特征向量的复杂工作，但其隐式计算容易产生噪声，计算复杂度较高，其性能虽超过了基于人工构造特征向量的方法，但其运算、训练速度较慢，面对大规模文本时仍有不足。

传统机器学习的关系抽取方法选择的特征向量依赖于人工完成，且存在特征提取过程中产生误差传播，文本特征已被提取完全的问题，很大程度上限制了模型性能的提升，而基于核函数的特征向量方法存在计算易产生噪声，复杂度高，训练慢等缺点。愈来愈多的人研究人员，把视野转向深度学习，深度学习的关系抽取方法通过大量数据训练神经网络，自学习高阶、隐式语义特征，且训练出的实体关系抽取模型的精度较高。因此，基于深度学习的方法逐渐成为了实体关系抽取领域的研究热点^[9]。

1.2.2 深度学习方法

基于深度学习的实体关系抽取方法,根据实体识别和关系抽取两个子任务完成的先后顺序,可分为两类方法,分别为: Pipeline (流水线)方法和 Joint (联合抽取)方法。其中流水线学习方法是在先通过一个实体标注模型标注关系实体,后来在识别出的实体集合之中进行关系分类,联合学习方法是同时进行实体识别和实体关系抽取^[9]。

基于流水线的方法进行关系抽取的主要有:使用基于 RNN (循环神经网络), CNN (卷积神经网络), LSTM (长短时记忆网络) 及其改进模型的网络结构。

基于 RNN 进行关系抽取的方法由 Socher 等^[24]首次提出,此方法解决了单词向量空间模型无法捕捉到长短语构成意义的问题。Zeng 等^[25]首次使用 CNN 提取词级和句级特征,输入到 MLP (多层感知机) 和 Softmax 激活函数,输出关系概率,进而实现关系分类,显著提高了关系抽取模型的精度。Liu^[26]等使用 CNN 模型,将距离向量等原始数据结合进文本向量进行实体关系抽取。Nguyen 等^[27]在 CNN 中,设计使用了多种窗口大小的卷积核,能捕捉隐含特征,减轻了对外部工具包的依赖,和计算复杂度。

在中文研究方面,孙建东等^[28]基于 COAE2016 数据集,提出了一种用于中文实体关系的抽取的方法,有效结合了 SVM 和 CNN 算法。高丹等^[29]提出一种结合 CNN 和改进核函数的多实体关系抽取方法,取得了较优秀的抽取效果和计算效率。吴等人^[30]提出一种基于神经元块级别注意力机制的关系抽取模型。Xu 等人^[31]提出一种结合 LSTM 和句法依存分析树的最短路径的实体关系抽取方法,且融合了词性、词义、句法等特征。Zhang 等人^[32]使用了 BiLSTM 神经网络,结合词与词之间的交互信息,进行关系抽取。Zhong 等人^[33]采用片段级别实体识别,并在实体间插入特殊关系符号,进行实体识别和关系抽取,并取得优秀结果。

流水线方法虽然成果斐然,但模型易产生误差传播。实体识别模块中的误差将传递到关系抽取模块中,错误或无关的实体进行关系配对时会产生冗余信息降低模型性能。此外,流水线式的处理方式忽视这两个子任务之间的内在依赖和联系,无法充分利用输入信息。相比于流水线方法,联合学习方法能够同时抽取实体和实体关系,并且充分利用实体和关系间紧密的交互信息和前后约束,能完善处理流水线方法所出现的问题^[9]。

目前,联合抽取的主流建模方式主要分为四种:

(1) 将实体关系抽取建模为端到端的表格填充 (Table-Filling)。Gupta 等人^[34]基于表格填充提出的关系抽取方法,尽管该方法让实体抽取模块和关系抽取模块共享参数,但实体和实体关系的分开提取,会产生冗余信息。

(2) 将实体关系抽取建模为序列标注 (Sequence-Labeling)。Zheng 等人^[35]将实体识别和关系识别统一为一个词元级别的序列标注任务, 由于每个词元只能分配一个标签, 尽管实现了单步抽取, 但无法克服实体嵌套问题。Dai 等人^[36]在此基础上采用多轮标注方法, 有效解决实体嵌套问题。这类基于标注的方法都需要人为设计精妙的标注体系。

(3) 将实体关系抽取建模为编码器-解码器 (Encoder-Decoder) 结构^[37,38]。这类方法在保证计算复杂度良好的情况下能有效应对关系重叠问题, 但较难解决暴露偏差和嵌套实体问题。

(4) 将实体关系抽取建模为主实体到客实体之间的关系映射。Wei 等人^[39]提出一种级联抽取框架先提取可能涉及目标关系的候选主实体, 然后为每个提取的主实体标记相应的对象关系和客实体。

此外, Miwa 等人^[40]尝试使用基于共享参数的联合抽取方法, 使用了句法依存树和 LSTM 来编码实体, 仍会造成实体冗余。Bekoulis 等人^[41]将关系抽取视为一个多头选择问题, 同时使用标签嵌入融入标签特征, 但解码的是词元级别关系, 需要联合实体识别模块识别出的实体边界来共同解码实体关系。Katiyar 等人^[42]使用指针网络来解码实体关系, 存在对客实体的遗漏, 且没有充分利用实体边界信息, 易导致注意力机制分散。Li 等人^[43]将关系抽取转化为多轮对话任务, 可以很好捕捉层级间的依赖关系, 设计好的问题包含了重要的先验关系信息, 采用了强化学习机制来缓解误差积累的问题。Dixit 等人^[44]使用片段排列和片段筛选方式来生成候选片段, 但最后进行片段关系分类时会产生冗余, 导致精度下降。Fu 等人^[45]提出基于图卷积神经网络 (Graph Convolutional Network, GCN) 端到端的联合抽取模型, 通过关系加权 GCN 来考虑命名实体和关系之间的交互, 从而更好地提取关系。Wang 等人^[46]提出一种基于握手标注和词元链接方法, 虽然单步解码出实体关系三元组, 却仍然需要结合多个词元的识别结果来进行关系解码。Sui 等人^[47]将联合实体和关系提取视为一个集合预测问题, 使用了非自回归解码器, 且使用了二部图匹配损失函数, 但受预先确定的三元组数量超参数所限制。

综上所述, 联合抽取方法较流水线式抽取方法取得较大进步, 但还是难以解决暴露偏差、实体嵌套、关系重叠问题, 仍有较大的探索空间。

1.2.3 存在问题

目前中文关系抽取模型中普遍存在以下问题:

(1) 实体嵌套: 一个较长的实体中包含若干个较短实体, 模型进行实体识别时

无法识别所有实体，如“长春西站”属于机构实体，包含“长春”，“长春”属于地名实体；“长春市宽城区”属于地名实体，包含“长春市”和“宽城区”两个地名实体，当前大部分模型只能识别出其中之一，不能识别出所有实体，进而对后续的关系识别造成影响。

(2) 暴露偏差：由于模型训练时使用真实标签，在预测时使用模型生成的标签，而这两种标签分布不一致从而产生的误差问题，这种问题通常还伴随着误差积累问题。

(3) 关系重叠：一个句子中的不同关系三元组间存在某些相同实体，模型无法识别出所有的存在的实体关系三元组。文本按照关系重叠类型可分为三类：常规类型（Normal）、实体对关系重叠（Entity Pair Overlap, EPO）和单实体关系重叠（Single Entity Overlap, SEO）。Normal 型为文本中只存在一种实体关系；EPO 型为实体对关系重叠，即一个实体对之间存在着多种关系；SEO 型为单实体关系重叠，即单个实体与其他不同实体存在关系。如图 1.1 所示。

这三类问题，对关系抽取模型性能有着显著影响，且较难解决，解决这三类问题，是十分迫切且具有相当价值的。

关系重叠类型	文本	三元组
Normal	奥巴马出生于美国。	(奥巴马, 出生地, 美国)
SEO(Single Entity Overlap)	奥巴马出生于美国，毕业于哈佛大学。	(奥巴马, 出生地, 美国) (奥巴马, 毕业学校, 哈佛大学)
EPO(Entity pair Overlap)	周星驰自导自演《功夫》。	(周星驰, 导演, 《功夫》) (周星驰, 主演, 《功夫》)

图 1.1 关系重叠类型图

1.3 本文主要工作

本文针对关系抽取技术存在的实体嵌套、暴露偏差，关系重叠问题展开研究，采取基于片段的实体关系抽取思路，设计了词元（Token）序列到片段（Span）序列的标记，将关系抽取问题转化为多头片段关系分类和片段实体关系标注问题，构建了两个基于片段的实体关系联合抽取模型，本文的具体主要工作如下

(1) 提出了一种基于词元的片段标记方法，将固定窗口长度内的组成片段的头尾词元位置索引组合成一个索引元组，枚举出所有窗口内的片段，进行关系抽取，能完善解决实体嵌套问题。此外，还提出了三种不同的片段映射策略将词元序列重新平

铺成片段序列，为后续进行片段嵌入、片段特征提取和片段关系抽取构建基础。

(2) 结合上述标记方法，提出两个关系抽取模型。

1) 基于片段多头选择的实体关系联合抽取模型 (Span based Multi Head Selection, SMHS), 首先通过片段嵌入方式直接构造片段语义向量, 然后使用 LSTM 和多头自注意力机制进行片段深层特征提取, 最后采取多头选择方法, 单步解码出实体关系三元组, 解决关系重叠问题的同时, 也解决了暴露偏差和误差积累问题。

2) 基于片段标注的实体关系联合抽取模型 (Span-Labeling Based Model, SLM), 设计了特殊的实体关系标签, 首先, 同样通过片段嵌入的方式构造片段语义向量, 然后使用 GRU 和多头自注意力机制进行片段深层特征提取, 最后采取多层标注方法进行关系标签分类, 结合标签关系单步解码出关系三元组, 缓解了关系重叠问题, 解决暴露偏差问题。

(3) 在由百度公司出品的权威公开中文数据集 DuIE2.0 上进行实验, 包括: 对比实验、消融实验、影响因素实验。对比实验相较于各类基线模型取得了更优秀的抽取结果, 验证了所提出模型的有效性; 消融实验较没有加入所设计模块的模型, 取得了更优秀的结果, 验证所设计模块的有效性; 影响因素实验通过控制变量的方式, 确认了潜在实验参数对模型性能的影响。

1.4 本文章节架构

本文主要内容分为 5 个章节, 具体如下:

第一章: 绪论, 介绍了关系抽取研究背景和意义; 综述了关系抽取的研究现状, 总结了几类关系抽取方法的优缺点; 说明了本文的主要研究工作; 介绍了本文的整体架构。

第二章: 相关技术基础, 介绍了研究过程中所使用的主要相关技术、理论, 包括预训练模型, 神经网络模型, 注意力机制, 本文所提出的片段标记和片段映射方法。

第三章: 基于片段多头选择的实体关系联合抽取模型 (SMHS), 主要说明所提出的 SMHS 细节。SMHS 将关系抽取问题视为片段多头关系分类问题, 将由预训练模型 BERT 生成的词元向量构造出片段向量, 然后输入到片段特征提取模块, 最后通过段类型分类器和片段多头关系分类器抽取出实体和实体关系。内容包括: 模型架构, 片段编码方式, 关系解码方式, 损失函数, 实验结果及分析。

第四章: 基于片段标注的实体关系联合抽取模型 (SLM), 主要说明所提出的 SLM 细节。SLM 将关系抽取问题视为片段标注问题, 将由预训练模型 ALBERT 生成的词元向量构造出片段向量, 然后输入到片段特征提取模块, 最后通过片段关系标签分类

器,标注上实体关系标签,通过进行实体关系标签匹配,抽取出实体关系。内容包括:模型架构,实体关系标签设计,片段编码方式,关系解码方式,损失函数,实验结果及分析。

第五章:总结与展望,总结文章主要工作结果,进一步探讨和展望未来研究重点和研究方向。

最后是致谢和参考文献等部分。

第 2 章 相关技术基础

2.1 预训练模型

篇章文本是由若干句子组成，而句子是由一个个汉字组成，所以如何更好地将汉字表示成计算机可以理解的向量表示，更好地表达汉字中所蕴含的文本语义信息，是研究人员一直在探究的问题。预训练模型技术应运而生，其主要思路是，运用大型语料预训练出相关汉字的字词向量表示，将具有丰富语义信息的字词向量，接入到相关下游 NLP 任务中，以此来提高相关 NLP 任务的指标。预训练模型技术在当前的自然语言处理技术占有举重若轻的地位，在几乎所有主流的 NLP 任务都取得了成功。以下将简略介绍几种静态与动态预训练模型。

2.1.1 静态词嵌入预训练模型

独热编码 (One-Hot)，研究人员最初使用的方法，就是一个单词用长度为 V 的向量表示，其中只有一个位置为 1，其余位置为 0， V 为语料中词库的大小。假设从语料库中选出若干个词作为词表，大小为 4，“长春”这个名词，在词表中的序号为 1，则其向量表示为 $[1,0,0,0]$ ，同理“大学”这个名词，在词表中的序号为 2，则其向量表示为 $[0,1,0,0]$ 。虽然能让分类器处理离散数据，但当词表数量过大时，会造成维度爆炸和向量稀疏问题，无法捕获语义特征，无法表达不同语境下词语的不同含义，且词和词之间缺乏相关性。

Word2vec 模型^[48]，是 Google 在 2013 提出的一种基于 NNLM^[49]模型改进的词向量表示模型，包含两种训练模式，Skip-gram 和 CBOW。CBOW 是用目标词的上下文来预测目标词，输入是目标词固定窗口上下文所对应的 One-Hot 编码，与权重矩阵相乘在权重矩阵中，只有该词对应的 One-Hot 编码中为 1 的位置的权重才被激活，这个权重向量就是当前词对应的词嵌入。Skip-gram 原理相同，只是输入与输出相反，利用目标词预测固定窗口的上下文。Word2Vec 模型利用霍夫曼树代替了神经网络中间层到 Softmax 层的映射，降低了复杂度，且引入了负采样算法，语料库中出现的次数较多的词会更靠近树的根节点。Word2vec 虽然可以有效捕获词的语义特征，但也无法表达不同语境下的不同含义和解决一词多义等问题，用于不同下游任务时效果不佳，可迁移性较差。

2.1.2 动态词嵌入预训练模型

(1) ELMo

ELMo (Embeddings from Language Models) ^[50] 是一种基于大规模语料特征抽取式，双向自回归语言预训练模型，其基本架构如图 2.1 所示。其以 LSTM (Long Short-Term Memory) 为基本结构，通过在输入层和输出层使用 CNN (Convolutional Neural Network)，减少了词表规模，解决了未登录词的问题；底层使用静态词嵌入，而后使用双层 LSTM 结构捕获上下文、句法、语义信息，能有效解决一词多义问题，然后接入下游 NLP 任务，进行动态调整。

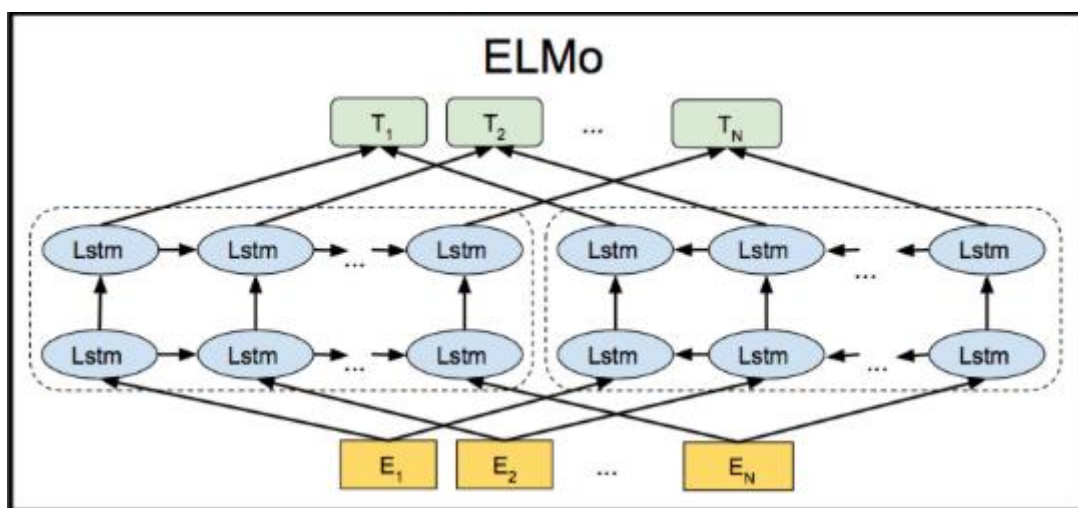


图 2.1 ELMo 基本结构图

(2) BERT

BERT (Bidirectional Encoder Representation from Transformers) ^[51] 是 Google 2018 年提出的基于 Transformer 的降噪自编码语言模型，通过堆叠 24\12 层的 Transformer 结构作为编码器和解码器以此为基本架构，层与层之间采用残差连接和归一化处理，基本结构如图 2.2 所示。

BERT 的预训练任务创新性地使用了 MLM (Masked Language Model) 和 NSP (Next Sentence Predict)。MLM 训练任务类似英语试题中的完形填空，具体做法为：训练时，将输入序列随机选择 15% 用 [MASK] 标签进行替换，模型通过上下文来预测被 [MASK] 标签掩盖的词，而其中被选中的词再选择 10% 保持不变，10% 替换成随机词，来引入噪声，增强模型的鲁棒性。NSP 任务是为了模型更好理解多个句子间的关系，以适应自然语言生成、问答系统等 NLP 任务，其做法为：训练时，选择 50% 有

前后关系的句子对，50%随机选择，在句子对的第一个句子前插入[CLS]标签来作为句子的向量表示，两个句子末尾插入[SEP]标签，输入到模型之后，使用 Softmax 层进行二分类，来判断句子对之间是否有关联关系。采用多任务学习的方式，两项训练任务在训练时一同进行，损失函数为两项任务损失函数的组合。

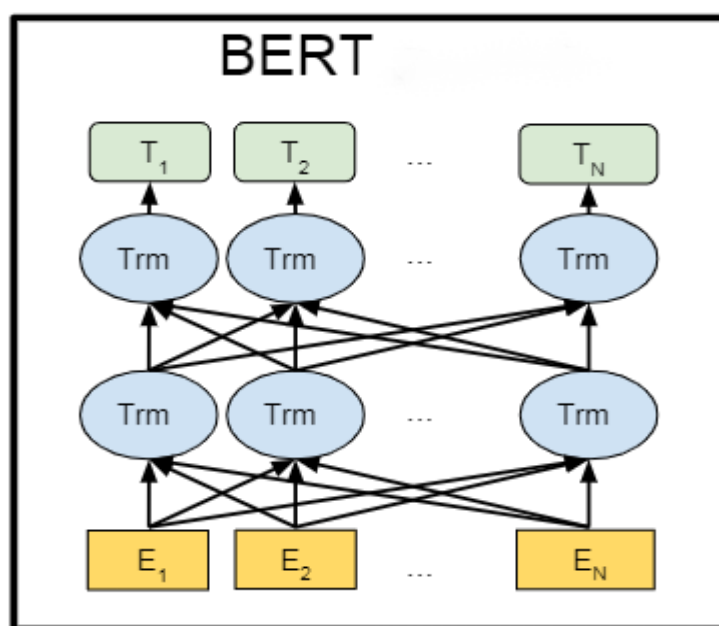


图 2.2 BERT 基本结构图

基于 BERT 的自然语言处理任务主要分为两步，首先使用 BERT 在大规模语料库上进行预训练，直至模型参数收敛，模型充分学习了大量语义信息，形成有效的文本向量表示，然后将预训练完成的 BERT 作为起始模型，接入到其余下游 NLP 任务，进行整体模型的微调（Fine-Tuning）或者固定（Frozen）BERT 模型参数，只对下游模型结构参数进行训练。这种方式在面对数据集规模较小时也能依靠 BERT 含有的大量语义信息获得不错的效果。

BERT 的出现具有里程碑式的意义，极大地推动了 NLP 领域的发展，基于“预训练-微调”的方法已经成为 NLP 研究的主流，任何下游 NLP 任务，均可将训练好的 BERT 或者基于 BERT 的改进预训练模型作为组件使用，节省了从头开始训练的时间和资源。

（3）ALBERT

ALBERT（Alite BERT，ALBERT）^[64]，顾名思义一个精简的 BERT。由于 BERT 模型的参数量过于庞大，常常包含数亿甚至数十亿参数，在实际使用的过程中很容易

受到硬件内存的限制模型训练时间也过于漫长, ALBERT 的提出就是为了解决上述问题。ALBERT 模型的结构还是依照了 BERT 的框架, 采用了 Transformer 以及 GELU 激活函数。其主要采用了两种方式削减了参数量: 一是因式分解嵌入层的参数。分解庞大的词汇嵌入矩阵为较小的矩阵, 从而将隐藏层的大小与词汇嵌入的大小分离开来。这种分离使得隐藏层的增加更加容易, 同时不显著增加词汇嵌入的参数量。二是跨层的参数共享, 避免参数量随着网络深度的增加而增加, 同时不对其性能造成显著影响, 提升参数效率, 提升训练速度。这些参数削减技术还可以充当某种形式的正则化, 可以使参数训练更加稳定, 而且有利于泛化。ALBERT 还采用了新的训练任务 SOP (Sentence Order Prediction), SOP 任务是句子顺序预测, 能显著提升下游多句子编码任务的性能, 同时移除了 dropout。这些手段都有益于提升训练速度, 简化参数量, 进而使预训练模型的应用范围更加广泛。

2.2 神经网络模型

2.2.1 循环神经网络 (RNN)

由于传统的神经网络中, 模型不会关注上一时刻的已经处理过的信息, 只关注当前信息。由于传统神经网络缺少记忆功能, 但在类似自然语言处理等领域中处理的序列多是有前后信息交互, 基于对处理时序序列的强烈需求, 循环神经网络便应运而生。

循环神经网络 (RNN) 是一类以序列数据为输入, 在序列的演进方向进行递归, 且所有节点(循环单元)按链式连接的神经网络^[52], 能记忆当前节点之前的输入信息, 并且利用这些信息进行后续节点的输出计算, 从而达到前后信息交互。当前 RNN 已经在自然语言处理、图像识别、语音识别等领域迅速取得了巨大成功以及广泛应用。一个简单的循环神经网络结构的隐藏层, 不仅包括输入层的输出还有来自上一节点隐藏层的输出, 具体结构如图 2.3 所示。

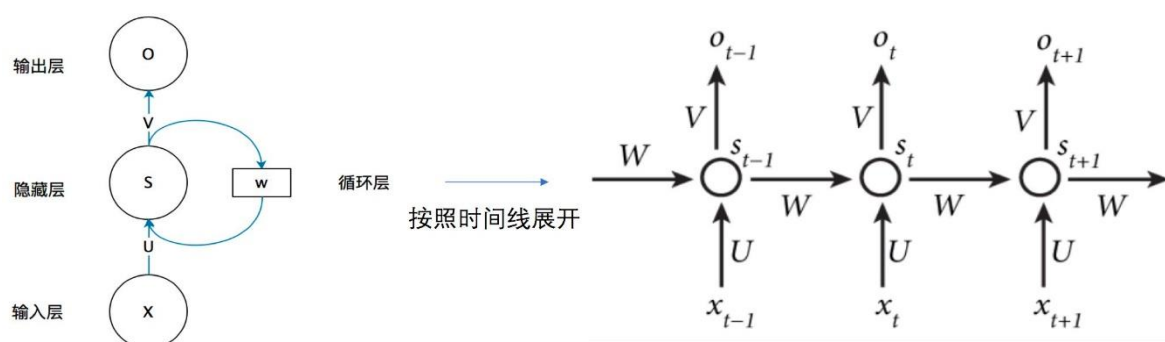


图 2.3 RNN 基本结构图

图中， X_t 为 t 时刻输入的值， O_t 为 t 时刻隐藏层的输出值，一般为 $O_t = \text{softmax}(VS_t)$ ，或者使用其他的激活函数，最后可得到 S_t 的计算公式，如公式 2.1 所示。

$$S_t = f(UX_t + WS_{t-1}) \quad 2.1$$

其中， W 为上一时刻的记忆权重矩阵， U 为此时输入值的权重矩阵， V 是输出值的权重矩阵，一般来说三者共享权重。RNN 还有许多变形，如：一对一，一对多，多对一，局部多对多，全部多对多等，如图 2.4 所示。

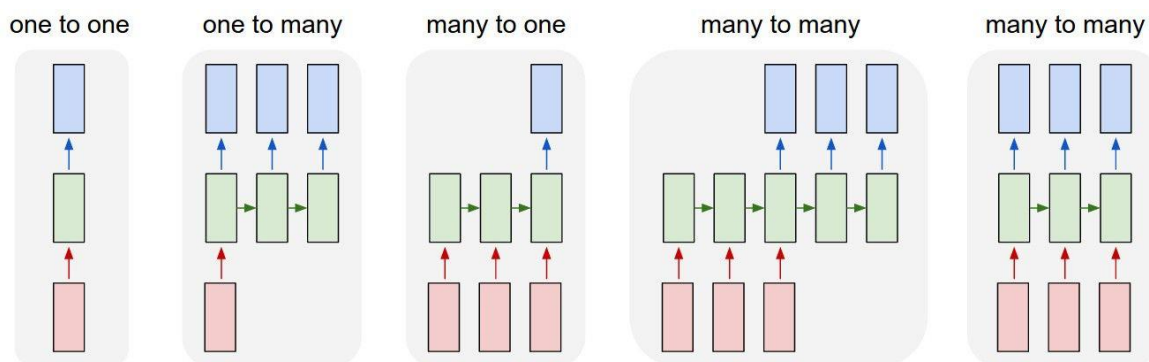


图 2.4 RNN 多结构图

虽然 RNN 能够有效处理变长的时序文本，但在实际的模型训练过程中存在着“梯度消失”、“梯度爆炸”和“长距离依赖”等问题。梯度爆炸、消失问题是指，随着文本的长度增加，计算复杂度大幅提高，计算公式的反向梯度，会趋向于零或者一直变大，权重矩阵无法更新；长距离依赖问题是指，距离较远的文本信息经过较长时间的计算后特征被覆盖，丧失了学习较远文本信息的能力，导致模型的性能急剧下降^[53]。

2.2.2 长短期记忆网络（LSTM）

长短期记忆网络（Long Short-Term Memory, LSTM）是基于传统循环神经网络的改进网络，顾名思义它具有记忆长短期信息的能力，是为了解决传统神经网络存在“长距离依赖”、“梯度消失”、“梯度爆炸”等问题而专门设计^[54]。

LSTM 的关键就是细胞状态，水平线在图上方贯穿运行。细胞状态类似于传送带。信息直接在整个传输带上运行，只有一些少量的线性交互，信息不易收到干扰。LSTM 有通过精心设计的称作为“门”的结构来去除或者增加信息到细胞状态的能力；通过引入门（gate）机制用于控制特征信息的流通和损失，包含一个 sigmoid 神经网络

络层和一个 pointwise 乘法操作，相较于普通的 RNN，LSTM 在长序列文本中具有更好的表现。

LSTM 的门机制分为：输入门（input gate）、遗忘门（forget gate）、输出门（output gate），三种门机制类似三种信息传递开关，如图 2.5 所示。

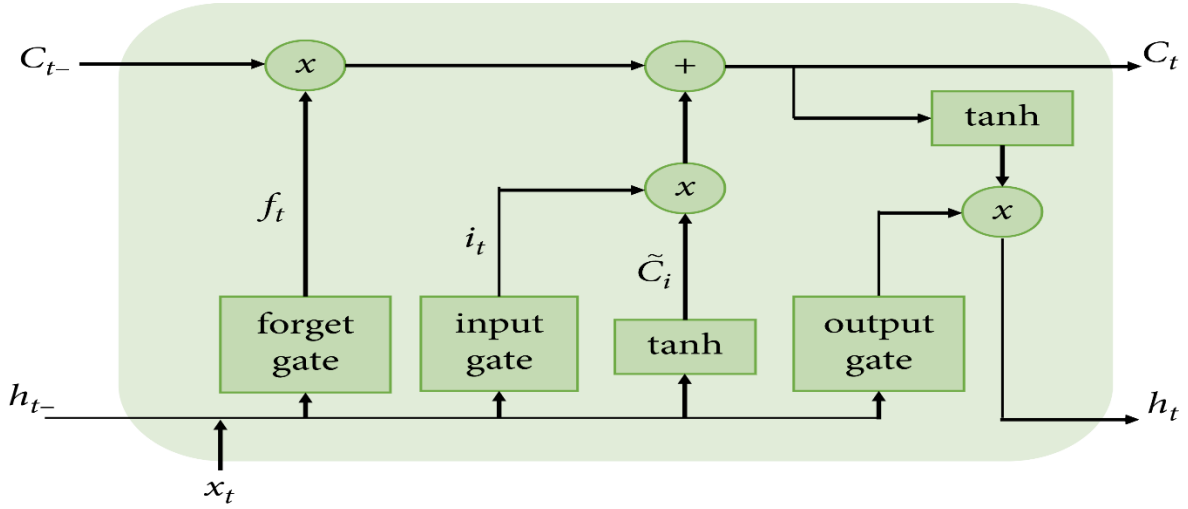


图 2.5 LSTM 基本结构图

输入门对当前时刻输入 x_t 和前一时刻隐藏层输出 h_{t-1} 对细胞状态进行控制。计算公式如公式 2.2 和 2.3 所示。

$$i_t = \sigma(W^i x_i + U^i h_{t-1} + b_i) \quad 2.2$$

$$\tilde{c}_t = \tanh(W^c x_i + U^c h_{t-1} + b_c) \quad 2.3$$

遗忘门对前一时刻的细胞状态 c_{t-1} 状态进行控制，计算公式如公式 2.4 所示。

$$f_t = \sigma(W^f x_t + U^f h_{t-1} + b_f) \quad 2.4$$

输出门对当前时刻细胞状态进行控制，计算公式如公式 2.5 所示。

$$o_t = \sigma(W^o x_t + U^o h_{t-1} + b_o) \quad 2.5$$

当前时刻的细胞状态输出由遗忘门的输出 f_t 和输入门的输出 i_t 控制，其计算公式

如公式 2.6 所示。

$$c_t = f_t \cdot c_{t-1} + i_t \cdot \bar{c}_t \quad 2.6$$

当前时刻 t 的隐藏状态输出计算公式如公式 2.7 所示。

$$h_t = o_t \cdot \tanh(c_t) \quad 2.7$$

公式 2.2-2.7 中涉及的 W 、 U 和 b ，分别为权重矩阵和偏置量，通过模型训练学习得到， \cdot 表示为矩阵点乘， σ 为 *sigmoid* 激活函数。

2.2.3 门控循环单元（GRU）

门控循环单元（Gate Recurrent Unit，GRU）是基于 LSTM 进行的一种改进神经网络。GRU 与 LSTM 都能解决“长距离依赖”、“梯度爆炸”等问题，其设计的主要目的是针对 LSTM 神经网络计算复杂度高的问题。

GRU 与 LSTM 不同的是，GRU 只有两个门控结构，称为重置门（reset gate）和更新门（update gate），没有细胞状态，网络结构较 LSTM 简化，模型训练过程中所涉及到的参数量、计算复杂度降低，训练速度加快，模型性能和 LSTM 比相差无几，其基本结构如图 2.6 所示。

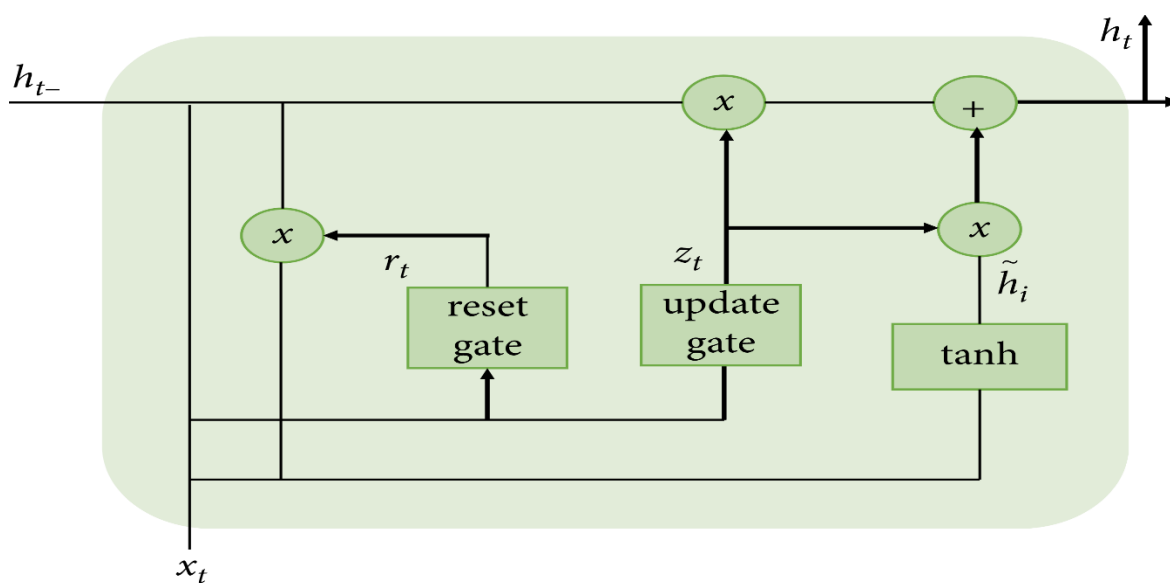


图 2.6 GRU 基本结构图

重置门对当前时刻的输入信息和对前一时刻的状态信息如何结合进行控制，其计算公式如 2.8 所示。

$$z_t = \sigma(W^z x_i + U^z h_{t-1} + b_z) \quad 2.8$$

更新门对前面记忆信息保存到当前时刻的量进行控制，其计算公式如公式 2.9 所示。

$$r_t = \sigma(W^r x_i + U^r h_{t-1} + b_r) \quad 2.9$$

候选隐藏状态通过前一时刻的隐藏状态 h_{t-1} 和重置门计算得到，其计算公式如公式 2.10 所示：

$$\bar{h}_t = \tanh(W^h[x_t, h_{t-1} \cdot r_t]) \quad 2.10$$

当前时刻 t 的隐藏状态输出计算公式如公式 2.11 所示。

$$h_t = (1 - z_t) \cdot h_{t-1} + z_t \cdot \bar{h}_t \quad 2.11$$

公式 2.8-2.11 中涉及的 W ， U 和 b ，分别为权重矩阵和偏置量，通过模型训练学习得到， \cdot 表示为矩阵点乘， σ 为sigmoid激活函数。

2.3 多头自注意力机制

注意力机制（Attention Mechanism）的灵感起源于人类的选择性注意，人类可以通过快速扫描图片或文本，确定需要重点关注的区域，比如一篇文章的首段，尾段；一段话的中心句等等，然后投入更多的注意力，以获取需要的信息，抑制其他无用的干扰信息。在诸多领域都取得了成功和广泛的应用。对于自然语言处理来说，就是根据文本向量所含语义信息的重要程度，赋予不同的权重系数，越是重要的信息权重越大，无需像传统机器学习手动进行特征工程工作，由神经网络自动提取特征信息，从而帮助模型提升性能。

注意力机制的实质计算查询向量对键向量权重分数的过程，给定一个任务相关的查询（Query）向量 Q 通过计算与键（Key）向量 K 的注意力分布附加到值（Value）向量 V 上，从而计算出注意力的值。整个过程大致可以分为三步：一是输入信息向量，输入查询、键向量；二是通过不同的注意力打分公式，计算注意力的分布；三是根据注意力分布来计算输入信息的加权平均，进而得到输入信息的权重值。

多头自注意力机制（Multi-head Self-attention Mechanism）则是传统注意力机制的变种，能够对输入序列建立长距离依赖关系，也能通过 mask 机制来处理变长序列。其和传统注意力机制不同之处在于，传统的注意力机制中的查询向量 Q 是通过外部给出的，而键向量 K 和值向量 V 一般与输入的文本向量 X 相等，而多头自注意力机制中的 Q , K , V ，三个向量是通过与三个动态的权重矩阵与输入的文本向量 X 进行线性变换得到的；多头机制则是利用多个查询向量并行地从输入信息中提取到多组不同信息进行拼接，在不同位置共同关注来自不同表示子空间的信息，有助于捕捉到更丰富的特征信息，如图 2.7 所示。

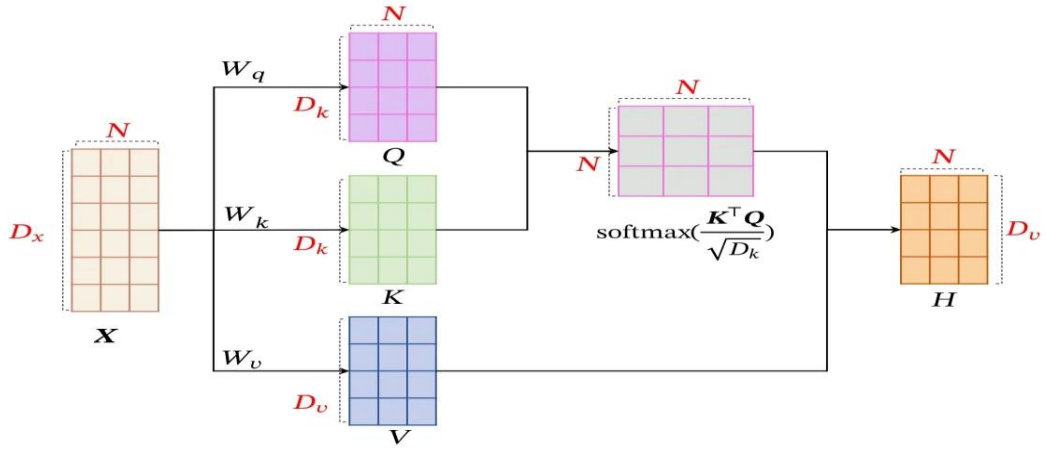


图 2.7 自注意力机制结构图

假设 $T = \{t_1, t_2, t_3, \dots, t_n\}$ 表示输入的文本向量 $X = \{x_1, x_2, x_3, \dots, x_n\}$ 经过多头自注意力机制计算后生成的语义向量，使用 z 个平行头进行特征捕获，其中第 i 个头的计算公式如公式 2.12 所示， T 的计算公式如 2.13 所示。

$$Head_i = \text{softmax} \left(\frac{(QW_i^Q)(KW_i^K)}{\sqrt{\frac{d}{z}}} \right) (VW_i^V) \quad 2.12$$

$$T = \text{Concat}(head_1, head_2, \dots, head_z) W^o \quad 2.13$$

其中 W_i^Q, W_i^K, W_i^V , W^o 是可训练的参数矩阵。

2.4 片段标记和片段映射策略

一般来说，在中文自然语言处理任务中很少使用整个词作为词元（Token）分句，是由于要引入分词工具进行预先分词，不可避免地会引入分词误差，且需要预处理，这和现在流行的端到端直接处理的思路相悖。

目前，主流的中文预训练模型都是以单字作为词元对文本进行分句，虽然避免了分词错误却导致了各种基于词元标注的序列标注模型只能预测单个字符的标签，识别实体时需要联合多个字符的识别结果，实体边界模糊，而且较难解决实体嵌套问题。由此，出现了基于片段进行实体关系抽取的思路。

片段（Span）是由若干个连续词元组合成的词元排列。按照文本语序进行枚举词元序列中的片段，让所有候选实体出现在所枚举的片段中，然后使用实体分类器进行分类，能单步抽取且不会遗漏该要抽取的实体，且不存在识别实体边界模糊的缺点，故能较为完善解决实体嵌套、误差积累问题。由此，需要将词元序列通过标记然后重新组合排列平铺成片段序列，本文设计了相应的词元转化为片段的标记和将标记好的词元序列转化为片段序列的映射策略。

片段标记的本质是利用词元位置索引来标记片段与其在片段嵌入矩阵中位置索引。例如：例如：“奥巴马在美国出生。”，其中“奥”标记为（0，0），“奥巴”标记为（0，1），“奥巴马”标记为（0，2）...，“美国”被标记为（6，7）...，依次类推。具体过程如图 2.8 所示。

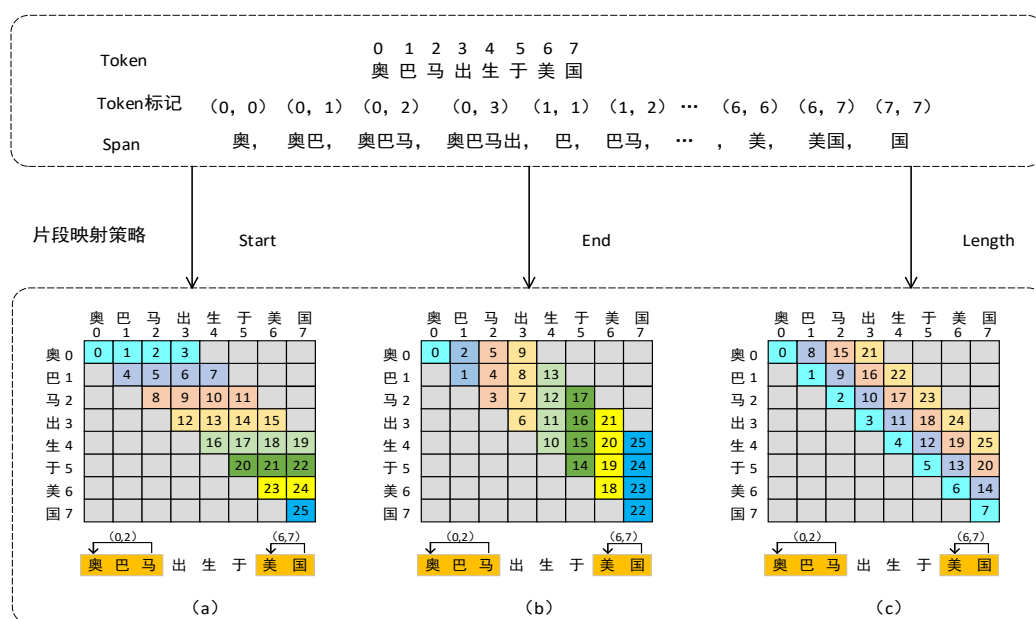


图 2.8 片段标记、片段映射策略图

片段映射是被片段标记之后的词元组合,通过不同的映射策略得到的词元组合平铺成新的片段序列中的位置索引。本文设计了三种映射策略,不同的映射策略有不同的应用场景,分别为:

(1) 片段相同起点的映射策略 (**start**): 固定文本片段的起点,通过更改片段的终点,即滑动窗口右边框来改变片段长度,直到达到窗口最大长度,将相同起点的片段进行集中排序,如图 2.8 中 (a) 所示, **start** 策略注重头词元的语义信息。

(2) 片段相同终点的映射策略 (**end**): 固定的是文本片段的终点,通过更改片段的起点,即滑动窗口左边框来改变片段长度,直到达到窗口最大长度,将相同终点的片段进行集中排序,如图 2.8 中 (b) 所示, **end** 策略注重尾词元的语义信息。

(3) 片段相同长度的映射策略 (**length**): 固定窗口长度遍历文本,然后改变窗口长度重新遍历该文本,直到达到窗口最大长度,将具有相同长度的片段提取出来并进行集中排序,如图 2.8 中 (c) 所示, **length** 策略则是对相同长度的片段实体更加敏感。

三种不同的映射策略,各有侧重,具体使用过程中按照实际情况进行选择,一般而言, **start** 策略在中文文本中效果较好,可以推广到中文以外的语言种类。

假设,文本的输入序列的词元个数为 n ,产生的片段数量为 m ,考虑到文本的时序性和语义性,每个词元只能与位于自身之前的词元来构成片段,若枚举所有片段会导致包含大量无用片段、消耗大量内存空间和算力,故设置一个最大片段长度 w 来保留可能存在的实体片段,并以此来过滤无用片段。片段数量 m 的计算公式如 2.14 公式所示。

$$m = \frac{(2n + 1)w - w^2}{2} \quad 2.14$$

其中, $0 < w \leq n$ 。

构造一个片段头尾词元位置索引元组转换到对应片段嵌入矩阵位置索引的映射 D 。假设某一片段头词元的位置索引为 i ,尾词元的位置索引为 j ,所构成的词元位置索引元组为 (i, j) ,其在片段嵌入矩阵的位置索引为 k ,公式如 2.15 公式所示。

$$D((i, j)) = k \quad 2.15$$

其中, $0 < i \leq j \leq n, 0 \leq k < m$ 。

2.5 本章小结

本章主要介绍,论文所使用的相关技术基础理论,包括几种常见的预训练语言模

型的结构、原理，常用的 LSTM 和 GRU 神经网络结构和计算公式，多头自注意力机的结构和计算公式，与本文所提出模型息息相关的片段标记和片段映射策略的结构和计算公式。这些基本技术理论，为接下来具体模型的构建和实验建立了坚实的理论基础。

第3章 基于片段多头选择的实体关系联合抽取模型

3.1 模型概述

针对现有实体关系抽取方法中存在的实体嵌套、误差积累、暴露偏差以及关系重叠问题，本文提出了一种基于片段多头选择的实体关系联合抽取模型（Span based Multi Head Selection，SMHS），将实体关系抽取转化为片段级的多头选择问题。

SMHS 整体可分为两个层次：编码层，解码层。编码层分为词元编码和片段编码，词元编码使用了以单个字符为词元的 BERT 中文版本预训练模型，预训练模型输出的词元向量、句子向量包含大量的外部语义信息，片段编码则结合预训练模型产生的词元向量、句子向量和片段嵌入方法来生成初始片段向量，利用长短时记忆网络、多头自注意力机制两个模块来进行片段特征提取，能直接有效地构造出片段语义向量，两个编码模块使用片段标记和不同的策略来映射词元和对应该片段之间的位置关系；解码层运用多头选择机制，通过枚举片段对之间关系的方式，单步解码出实体关系三元组，避免了暴露偏差、关系重叠、误差积累，且引入了用来辅助训练的片段类型分类任务，隐式融合片段类型信息，进行多任务学习。模型整体架构如图 3.1 所示。

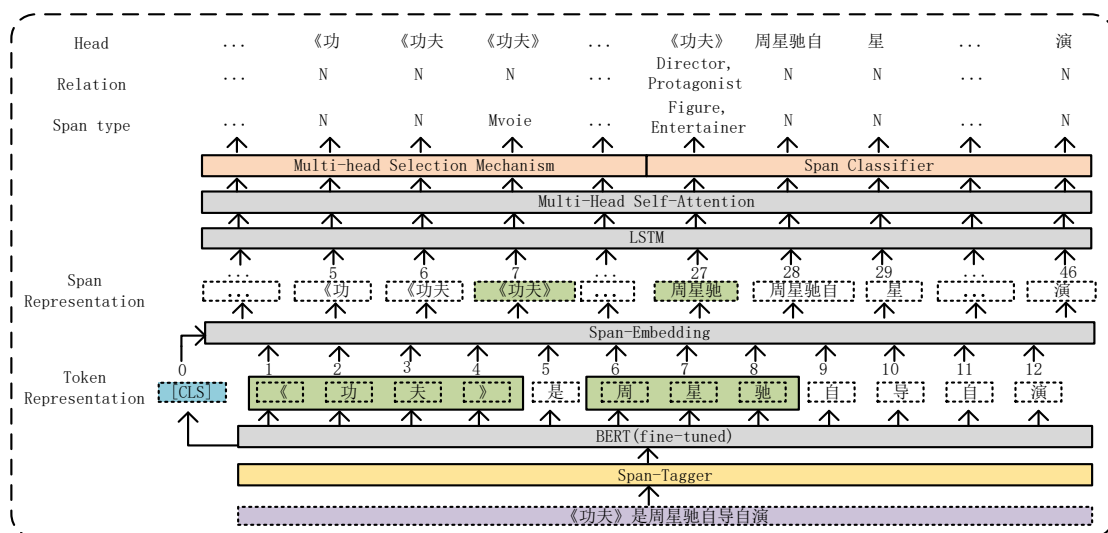


图 3.1 SMHS 模型架构图

其处理流程为：首先语料通过 BERT 预训练模型产生相应的词元向量和句子向量，其次将词元向量与句子向量送入片段嵌入模块构造出全部片段语义向量，利用片段标记器将片段向量一一映射到片段嵌入矩阵中，再送入长短时记忆网络、多头自注意力

机制进行片段信息交互和片段特征提取，最后用片段类型分类器、片段多头选择进行片段类型分类和片段关系解码，最终解码出实体关系三元组。

3.2 编码层

编码层包括词元编码层和片段编码层。词元编码层通过 BERT 预训练模型获得初步的词元语义向量，然后通过不同的片段映射方式进行组合，然后通过 LSTM 和多头自注意力机制获得具有深层特征的片段语义向量。

3.2.1 词元编码层

选用中文字符级 BERT 预训练语言模型来对语句进行词元编码，首先根据所选择的中文 BERT 模型的词表对文本进行词元化，然后根据设计好的片段标记器和相应的片段映射策略，确定组合后的片段在片段序列中的索引。然后输入词元，提取每个词元的上下文语义信息和句子整体语义信息，关于片段标记和片段映射的相关内容在 2.4 章节已经有详细介绍，关于 BERT 的预训练模型相关内容在 2.1.2 章节已经有详细介绍。

假设句子输入表示为 $S = \{s_1, s_2, s_3, \dots, s_n\}$ ，则经过 BERT 编码后的向量表示如公式 3.1 所示。

$$H, \bar{H} = BERT(S) \quad 3.1$$

其中， $H = \{h_1, h_2, h_3, \dots, h_n\}, H \in \mathbb{R}^{n \times d}$ ，表示为每个字符被 BERT 编码后产生的词元向量， $\bar{H}, \bar{H} \in \mathbb{R}^d$ ，表示为整个句子被 BERT 编码后产生的句子向量。 n 为序列长度， d 为 BERT 隐藏状态的维度。

3.2.2 片段编码层

SMHS 通过将组成片段的每个词元向量进行平均池化后与 BERT 的句子向量 CLS 进行拼接的方式构造片段语义向量，能直接提取片段特征。

假设 $H_{i:j} = \{h_i, h_{i+1}, \dots, h_j\}, H_{i:j} \in \mathbb{R}^{(j-i+1) \times d}$ ，表示在窗口长度为 w 下，位置索引为 i 的头词元到位置索引为 j 的尾词元中间所有词元所组成的词元向量矩阵，则其所组成的片段语义向量计算公式如公式 3.2—3.5 所示。

$$k = D((i, j)), 0 < i \leq j \leq n, 0 \leq k < m \quad 3.2$$

$$H'_k = \text{Meanpool}(H_{i:j}), H'_k \in \mathbb{R}^d \quad 3.3$$

$$x'_k = \text{Concat}(H'_k, \bar{H}), x'_k \in \mathbb{R}^{2*d} \quad 3.4$$

$$x_k = \text{Tanh}(\text{Linear}(x'_k)), x_k \in \mathbb{R}^d \quad 3.5$$

其中 m 为在序列长度为 n ，窗口长度为 w 下所有片段数量，即重新排列后的片段序列的长度， k 为头词元位置索引 i 到尾词元位置索引 j 所组成片段在片段序列中的位置索引， x_k 表示所组成片段的片段语义向量。

对构造出的所有初始片段向量，循环进行公式 3.2 至公式 3.5 的过程，然后根据片段标记器的映射结果，按序组成片段序列。 $X = \{x_1, x_2, x_3, \dots, x_m\}, X \in \mathbb{R}^{m*d}$ ，表示为所有片段语义向量重新组合成的片段序列。

3.2.3 LSTM 和多头自注意力机制层

使用 LSTM 和多头自注意力机制加强片段信息交互和深层片段特征提取。

LSTM 通过输入门、遗忘门和输出门对输入内容以及记忆单元里存储的内容进行控制，形成对之前输入信息的记忆，能有效解决梯度爆炸、梯度消失问题。片段嵌入矩阵通过 LSTM 进行编码，能有效加强片段与片段之间的信息交互，捕获片段之间的依赖关系。

注意力机制可选择性地关注文本的重要信息，多头自注意力机制是注意力机制的一种变体，其中 Q (query), K (key), V (value) 三者相等，利用多次查询并行地从输入信息中提取到多组不同信息进行拼接，在不同位置共同关注来自不同表示子空间的信息，有助于模型捕捉到片段更丰富的特征。

LSTM 和多头自注意力的相关内容在 2.2 和 2.3 章节已经由详细介绍

假设 $P = \{p_1, p_2, p_3, \dots, p_m\}$ ，表示为经过 LSTM 编码后的片段向量，其计算公式如 3.6 所示。

$$P = \text{LSTM}(X), P \in \mathbb{R}^{m*d} \quad 3.6$$

假设 $T = \{t_1, t_2, t_3, \dots, t_m\}$ ，表示为经过多头注意力机制编码后的片段向量，且使用 z 个平行头来捕获特征，则其中第 i 个头的计算如 3.7 所示， T 的计算公式如 3.8 所

示.

$$Head_i = softmax\left(\frac{(QW_i^Q)(KW_i^K)}{\sqrt{d/z}}\right)(VW_i^V) \quad 3.7$$

$$T = Concat(head_1, head_2, \dots, head_z)W^o \quad 3.8$$

其中 $W_i^Q \in \mathbb{R}^{d \times \frac{d}{z}}$, $W_i^K \in \mathbb{R}^{d \times \frac{d}{z}}$, $W_i^V \in \mathbb{R}^{d \times \frac{d}{z}}$, $W^o \in \mathbb{R}^{d \times d}$, W^o 是可训练的参数矩阵。

3.3 解码层

包括多头关系选择层和片段分类层, 使用多头选择机制进行单步的多头关系抽取, 能抽取所有存在的实体关系, 克服关系重叠、暴露偏差等问题, 片段分类层进行片段实体识别并作为辅助训练, 进行多任务学习融入实体信息和实体关系约束来提高任务指标。解码方式及步骤如图 3.2 所示。

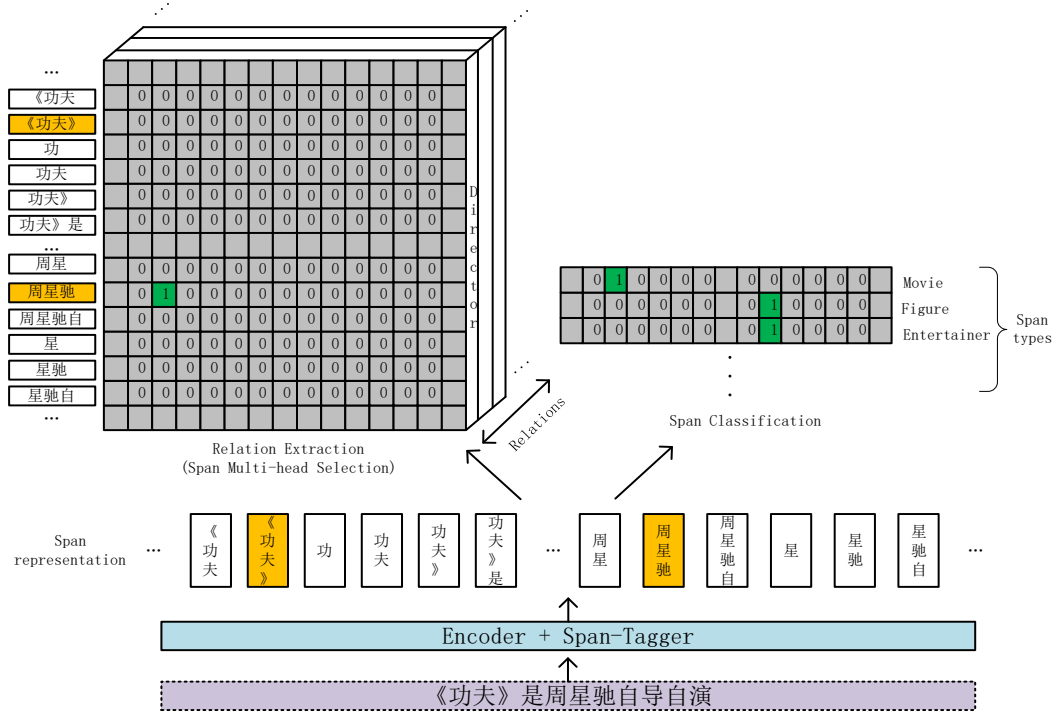


图 3.2 SMHS 解码步骤图

3.3.1 多头选择机制层

使用多头选择机制来进行关系抽取。以往多头选择机制是词元级，解码时会受到

实体识别结果影响而造成误差积累，以及在训练时使用真实实体标签，而测试时使用预测实体标签而造成的暴露偏差，而片段级多头选择直接在片段与片段之间进行多头关系选择，无需实体识别，单步抽取出主实体，客实体和关系，解决了积累误差的同时也解决了暴露偏差。

假设 $R = [r_1, r_2, \dots, r_l]$ ， R 表示为关系集合， l 为关系数量。 $\tilde{S}(t_i, t_j, r_k)$ 表示为主片段 t_i 与客片段 t_j ，两个片段间第 r_k 关系的得分。计算公式如公式 3.9 所示：

$$\tilde{S}(t_i, t_j, r_k) = V_k \tanh(U_k t_i + W_k t_j + b_k) \quad 3.9$$

其中 $V_k \in \mathbb{R}^d$, $U_k \in \mathbb{R}^{d \times d}$, $W_k \in \mathbb{R}^{d \times d}$, $b_k \in \mathbb{R}^d$ 。

最后计算主片段 t_i 与客片段 t_j ，是第 r_k 关系的概率：

$$P_r(\text{Head} = t_i, \text{Relation} = r_k | t_j) = \sigma(\tilde{S}(t_i, t_j, r_k)) \quad 3.10$$

其中 σ 为 sigmoid 函数。

3.3.2 片段分类层

实体信息尤其是实体类别信息被证明可以帮助提高关系抽取模型的效果^[57]，并且关系识别过程中存在实体类别约束，例如“歌手”关系只能由实体类型为“人物”和“歌曲”的实体来构成，为了利用实体类型、实体类型约束信息，模型通过加入片段类型分类任务共享片段编码进行多任务学习的方式来间接引入实体类型、实体类型约束信息，辅助片段关系抽取任务的训练。多任务学习^[58]可以利用多个任务之间的交互以及包含的特殊信息通过共享编码来提高模型的泛化性和鲁棒性。

假设 $I = [i_1, i_2, \dots, i_g]$ ， I 为片段类型集合， g 为片段类型数量。 $\bar{S}(t_i, i_k)$ 表示为片段 t_i 是第 i_k 类型的得分。计算公式如 3.11 所示。

$$\bar{S}(t_i, i_k) = Z_k * t_i + b_k \quad 3.11$$

其中 $Z_k \in \mathbb{R}^d$, $b_k \in \mathbb{R}^1$ 。

最后计算片段 t_i 是第 i_k 类型的概率：

$$P_i(\text{Type} = i_k | t_i) = \sigma(\bar{S}(t_i, i_k)) \quad 3.12$$

3.4 损失函数

对于片段多头选择的损失函数定义如下：

$$Loss_{re} = \sum_{i=1}^m \sum_{j=1}^m \sum_{k=1}^l -\log P_r(Head = t_i, Relation = r_k | t_j) \quad 3.13$$

对于片段类型分类的损失函数定义如下：

$$Loss_{sc} = \sum_{t=1}^m \sum_{k=1}^g -\log P_i(Type = i_k | t_i) \quad 3.14$$

最终的损失函数为：

$$Loss_{final} = \alpha * Loss_{re} + \beta * Loss_{sc} \quad 3.15$$

其中 α, β 为损失函数权重，用来平衡两个训练任务的主次关系。

训练时，使用AdamW优化器，Adam 优化器参数的更新不受梯度的伸缩变换影响，很适合应用于大规模的数据及参数的场景，适用于解决梯度稀疏或梯度存在很大噪声的问题^[60]，这两点都十分契合 SMHS 的训练。

3.5 实验与分析

3.5.1 数据集

数据集采用中文关系抽取数据集 DuIE2.0^[59]，是隶属于百度公司，业界规模最大的基于 schema 的中文关系抽取数据集，包含超过 43 万三元组数据、21 万中文句子及 48 个预定义的关系类型，43 个简单关系类型，5 个复杂关系类型。数据集中的句子来自百度百科、百度贴吧、百度信息流文本，数据标注通过人工标注和远程监督的方式生成。

DuIE2.0 数据集中有五种复杂关系类型。可以将数据集中的关系定义为：关系前缀+关系后缀。例如：“主演_@value”，关系前缀为“主演”，关系后缀为“_@value”。简单关系和复杂关系的区别为：在简单关系类型中，关系前缀唯一，关系后缀唯一，

而在复杂关系类型中，关系前缀唯一，而关系后缀不唯一，例如在配音关系中存在，配音_@value，和配音_inWork，两种关系的关系前缀相同，关系后缀不同。后缀为_@value 为此关系前缀大类的默认值，其他后缀为此关系前缀大类的附属关系。本文在处理复杂关系时，将关系前缀相同而关系后缀不同的关系视为不同的两种关系，故数据集共有 55 种关系。示例如表 3.1 所示：

表 3.1 关系示例表

简单关系		复杂关系
文本	《喜剧之王》主演周星驰。	王雪纯是 87 版《红楼梦》中晴雯的配音者。
三元组	{subject:《喜剧之王》， predicate: 主演_@value, object: 周星驰}	{subject:王雪纯, predicate: 配音_@value, object:《红楼梦》}, { subject:王雪纯, predicate: 配音_inWork, object: 晴雯}

由于构造文本片段和枚举片段与片段间的关系会耗费大量的硬件存储空间，受限于训练设备的原因，故将训练集和测试集中文本长度大于 62 及实体长度大于 15 窗口长度的文本剔除，构成了新的训练集与测试集。经过对比，新的训练集与测试集的关系类型分布基本与原数据集一致。后对筛选后的新数据集进行相关统计，统计信息包括各种关系重叠类型的文本数量（不重复统计，即包含多种重叠类型，只选择最复杂的重叠类型），包含不同实体关系三元组数的文本数量，数据集统计信息如表 3.2 所示，其中 N 表示单条文本中实体关系三元组的数量：

表 3.2 数据集信息表

实验数据	Normal	SEO	EPO	记录总数	三元组总数
训练集	59464	26999	3338	89801	144858
测试集	7208	3257	395	10860	17565
实验数据	N=1	N=2	N=3	N=4	N>=5
训练集	58584	18587	6440	3507	2683
测试集	7111	2238	717	445	349

原有的数据集标注方式并未标注出主实体、客实体在文本中的词元位置索引，所

以需要将数据集的标注方式进行转换,方便后续操作,由于 BERT 预训练模型的原因,在转换过程中添加 BERT 的特殊词元[CLS]和[SEP],故词元标记略有不同,具体示例如表 3.3 所示:

表 3.3 数据转换示例表

原标注数据	转换后标注数据
<pre> {"text": "《感恩父爱感恩母爱》是由朱自清编著的中国华侨出版社出版的图书", "spo_list": [{"predicate": "作者", "object_type": {"@value": "人物"}, "subject_type": "图书作品", "object": {"@value": "朱自清"}, "subject": "感恩父爱感恩母爱"}]} </pre>	<pre> {"text": "《感恩父爱感恩母爱》是由朱自清编著的中国华侨出版社出版的图书", "relation_list": [{"subject": "感恩父爱感恩母爱", "object": "朱自清", "predicate": "作者_@value", "subj_char_token_span": [2, 10], "obj_char_token_span": [13, 16], "subject_type": "图书作品", "object_type": "人物"}]}, "entity_list": [{"text": "朱自清", "type": "人物", "char_token_span": [13, 16]}, {"text": "感恩父爱感恩母爱", "type": "图书作品", "char_token_span": [2, 10]}]} </pre>
<pre> {"text": "《听赵教授讲家教的故事》是2007年石油工业出版社出版的图书,作者是赵忠心", "spo_list": [{"predicate": "作者", "object_type": {"@value": "人物"}, "subject_type": "图书作品", "object": {"@value": "赵忠心"}, "subject": "听赵教授讲家教的故事"}]} </pre>	<pre> {"text": "《听赵教授讲家教的故事》是2007年石油工业出版社出版的图书,作者是赵忠心", "relation_list": [{"subject": "听赵教授讲家教的故事", "object": "赵忠心", "predicate": "作者_@value", "subj_char_token_span": [2, 12], "obj_char_token_span": [35, 38], "subject_type": "图书作品", "object_type": "人物"}]}, "entity_list": [{"text": "赵忠心", "type": "人物", "char_token_span": [35, 38]}, {"text": "听赵教授讲家教的故事", "type": "图书作品", "char_token_span": [2, 12]}]} </pre>

3.5.2 实验环境与参数设置

实验采用 GPU 为 NVIDIA TITAN XP 显卡，每块显存 12GB，操作系统为 CentOS 7.9，运行内存为 128G，编程语言为 Python3.7，深度学习框架为 Pytorch1.8。在实际模型训练中，超参数值表如表 3.4 所示。

表 3.4 超参数表

超参数名称	超参数值
片段最大长度	15
词元最大长度	62
词元向量维度	768
批次大小	32
学习率	0.00001
最大迭代次数	200
α , β	0-1
注意力头数	12

3.5.3 实验评价标准

使用准确率 Precision、召回率 Recall 和 F1 值来评价实验结果，对于 SMHS 采用稍微宽松的标准：当主实体，客实体，关系类型判断正确就认为是正确的抽取结果，而不是严格的标准：当主实体，主实体类型，客实体，客实体类型，关系类型都判断正确才认为是正确的抽取结果。

计算公式如下：

$$Precision = \frac{TP}{TP + FP} \quad 3.16$$

$$Recall = \frac{TP}{TP + FN} \quad 3.17$$

$$F1 = \frac{2 * Precision * Recall}{Precision + Recall} \quad 3.18$$

其中 TP 为正确预测个数， FP 为错误预测个数， FN 为未被预测到的正确个数。

3.5.4 实验结果分析

为验证模型和辅助模块的有效性和寻找对模型性能有影响的因素，本文主要设计了三组实验：验证实验、消融实验、影响因素实验。验证实验选择了五个基线进行对比实验；消融实验针对是否加入片段类型分类任务、是否加入多头自注意力机制、片段类型分类任务共享的编码类型，设计相应实验；另外根据多任务学习中的两个超参数 α 和 β ；文本中三元组数量；文本关系重叠类型等因素对模型性能的影响进行实验。

(1) 验证实验

所采用的对比基线模型分别为：**CMAN**^[61]：标签信息与实体特征信息深度跨模态注意网络模型。**TP-Linker**^[46]：通过连接词元和词元标注的一个单阶段联合提取模型。**Novel-Tagging**^[35]：将关系抽取和实体提取两个任务转化为统一序列标注的模型。**CopyRE**^[62]：使用编码解码器架构，且采用了复制机制，可以有效提取重叠的关系三元组。**WDec**^[63]：**CopyRE**的改进模型，能够处理不同类型关系重叠和不同长度的实体。以上模型对句子进行词元编码时都使用相同的**BERT**预训练模型。实验结果如表3.5所示。

表 3.5 主要实验结果表

Model	Precision	Recall	F1
BERT+Novel-Tagging	0.758	0.645	0.697
BERT+CMAN	0.773	0.681	0.724
BERT+TP-Linker	0.784	0.705	0.742
BERT+CopyRE	0.762	0.695	0.727
BERT+WDec	0.764	0.699	0.730
BERT+SMHS	0.809	0.729	0.767

从表 3.5 可以看出，**SMHS** 在精准率，召回率，F1 值，均取得了最先进的水平。所使用的基线模型都基于词元向量进行解码，存在对实体边界识别不清晰的缺点。其中，**Novel-Tagging** 无法克服重叠实体、嵌套实体的问题，导致召回率偏低；**CMAN** 模型虽然运用跨模态的方法结合了标签信息，但本质上并未解决重叠实体、暴露偏差问题，导致 F1 指标比本文模型低 4.3%；**CopyRE** 和 **WDec** 均存在暴露偏差问题，对比同样单阶段解码的模型 **TP-Linker**，精准率提高了 2.5%，召回率提高了 2.4%，

F1 提高了 2.5%，是由于 TP-Linker 根据头尾词元结合的语义向量来进行分类，而片段之间的其他词元并没有参与，导致语义信息融合不充分且未利用类型信息，而本文模型将构成片段的所有词元向量和句子向量进行拼接融合，充分利用词元和句子信息，并利用辅助训练的片段类型分类任务融入片段类型信息，添加隐式的实体关系约束，较为完善地处理此类问题。

（2）消融实验

为了进一步探究所设计的方法的影响，进行了消融实验。实验内容分别为，去除了片段类型，去除了多头自注意力机制；使片段类型分类任务和关系抽取任务共享词元编码而不是共享片段编码。实验结果如表 3.6 所示。

表 3.6 消融实验结果表

研究内容	Precision	Recall	F1
去除 Span-Classify	0.770	0.694	0.730
去除 Multi Head Self-Attention	0.782	0.713	0.746
共享词元编码	0.791	0.707	0.747
正常模型	0.809	0.729	0.767

从表 3.6 可以看出，去除了片段分类任务导致 F1 降低了 3.7%，说明片段类型分类引入的片段类型信息对于关系抽取模型是有效的，是由于针对每种关系来说只有特定的类别的片段能够构成这种关系，而片段类型分类任务能够引入这种类型信息，对模型进行隐形的类型约束，进而提升模型性能；去除了多头自注意力机制导致模型的 F1 降低了 2.1%，说明多头注意力机制的片段特征提取对模型是有效的，是由于采用 LSTM 进行片段编码仅能加强片段之间的交互及依赖关系，无法对深层片段特征信息进行提取；不直接共享片段编码而是共享词元编码导致模型的 F1 降低了 2%，说明直接共享片段向量能更好利用片段类型信息、关系类型约束信息，是由于共享词元向量弱化了两个训练任务之间的联系，关系抽取模型无法直接有效获取片段类型信息。消融实验证明，模型中为提升关系抽取效果所设计的辅助模块和辅助训练任务是有效的。

3) 影响因素实验

为了更加清晰地探究，相关因素对模型性能影响，设计了多组影响因素实验，具

体有：多任务学习中的超参数 α 和 β 这两个损失权重参数对模型的影响程度；文本中三元组数量对模型的影响程度；文本关系重叠的类型对模型的影响程度；不同片段映射策略对模型的影响；不同的窗口长度对模型的影响。实验结果如表 3.7, 3.8, 3.9, 3.10, 3.11 所示。

表 3.7 损失权重实验结果表

编号	α	β	Precision	Recall	F1
1	0.3	0.7	0.761	0.702	0.730
2	0.4	0.6	0.769	0.709	0.738
3	0.5	0.5	0.782	0.708	0.743
4	0.6	0.4	0.789	0.713	0.749
5	0.7	0.3	0.809	0.729	0.767
6	0.9	0.1	0.794	0.716	0.753

表 3.8 三元组数量实验结果表

三元组数量 (N)	Precision	Recall	F1
N=1	0.851	0.756	0.800
N=2	0.802	0.743	0.771
N=3	0.788	0.707	0.747
N=4	0.809	0.702	0.752
N \geq 5	0.778	0.711	0.744

表 3.9 关系重叠实验结果表

关系重叠类型	Precision	Recall	F1
Normal	0.811	0.732	0.769
SEO	0.776	0.728	0.751
EPO	0.771	0.730	0.750

表 3.10 片段映射策略实验结果表

片段映射策略	Precision	Recall	F1
Start	0.809	0.729	0.767
End	0.801	0.728	0.762
Length	0.765	0.695	0.728

表 3.11 窗口长度实验结果表

窗口长度	Precision	Recall	F1
5	0.672	0.594	0.635
6	0.687	0.609	0.645
7	0.695	0.624	0.657
8	0.709	0.638	0.671
9	0.715	0.651	0.681
10	0.739	0.677	0.706
11	0.755	0.683	0.717
12	0.763	0.691	0.725
13	0.788	0.705	0.744
14	0.809	0.729	0.767
15	0.798	0.730	0.762

从 3.7 表中可以看出, α 和 β 的取值, 当 β 大于 α 时, SMHS 片段实体分类任务为住, 会削弱模型性能; 当 α 变大时, SMHS 效果会增强, 但过大时 SMHS 效果削弱, 是由于当 α 变大时, β 变小时, SMHS 参数进行训练时会以关系抽取任务的权重为主, 片段实体分类任务为辅助, 且使用融入片段实体信息, 进而增强模型; 但 α 过大时, SMHS 从片段实体识别任务中获取的片段实体信息不够充分, 模型学习不到潜在的实体关系约束, 导致 SMHS 效果下降。

从表 3.8 中可以看出, 随着文本三元组数量的上升, 文本语义复杂度急剧上升, 对模型的识别关系的干扰变大, 导致 SMHS 效果略微下降, 但下降的不多, 说明 SMHS 在面对复杂文本时也能较好地抽取出实体关系, 鲁棒性强, 抗干扰性强, 适合处理复杂文本, 适用性广泛。

从表 3.9 中可以看出, 面对不同关系重叠类型时, SMHS 效果也仅是略微下降, 这是由于, 所采用的多头片段关系选择机制, 通过对所有潜在的片段关系三元组进行打分, 进而筛选出关系, 不会遗漏该要抽取的实体关系三元组, 从根本上克服了关系重叠问题, 也进一步验证了 SMHS 的有效性。

从表 3.10 中可以看出, 使用 start 的映射策略时效果是最优秀的, end 映射策略次之, 而 length 策略是效果最差的。这是由于, start 策略通过共享头词元进而排列片段序列, 导致对头词元的语义信息较为敏感, 这和中文文本的特点所契合; end 策略则和 start 相反, 对于尾词元信息敏感, 适用的有效场景较少, 导致性能略有下降; length

策略只通过相同长度的片段进而排列，这对中文文本来说，会丧失较多的片段间的语义信息联系，导致模型效果下降显著。这也说明中文文本在进行实体关系抽取时，实体的头尾词元的语义信息较为重要且片段间的联系交互，对模型的性能均有较大影响，且头词元语义信息比尾词元语义信息更为重要。

从表 3.11 中可以看出，随着窗口长度的上升，SMHS 性能会增强。这是因为，当窗口长度较小时，无法识别出超出这个长度的实体，导致遗漏了所需要的实体且词元之间信息交互较少，窗口长度的增加，所能识别实体的长度范围变大，有益于模型的性能提升。数据集中最长的实体为 13 个词元，SMHS 在窗口长度为 14 时能取得最好效果，此时窗口长度再变大时，模型性能下降，是由于过长的窗口导致一个 span 中无关词元数量上升，其中杂糅了大量无关语义信息，损害了模型性能。但窗口长度上升会使 span 序列的长度急剧上升，使模型训练速度和推理速度急剧下降，在实际应用中，可以在窗口长度和推理速度间取得一个平衡的数值。

3.6 本章小结

本章主要是全面地介绍所提出的基于片段多头选择的联合抽取模型 SMHS 的相关内容，包括模型架构、模型处理步骤、计算公式、数据介绍、实验分析等。实验结果表明，所提出的 SMHS 比目前主流的模型效果都要好，面对复杂语境，三元组数量较多时，仍可以保持较好的性能，说明 SMHS 能有效解决实体嵌套、暴露偏差、关系重叠问题，验证了 SMHS 和辅助模块的有效性。

第 4 章 基于片段标注的实体关系联合抽取模型

4.1 模型概述

由于第三章所提出的 SMHS，存在时间空间复杂度较高，推理速度较慢等缺点，且现有基于标注的关系抽取模型，大多数是单层标注方法，很难解决文本中的实体嵌套、关系重叠和暴露偏差问题。为此，提出了一种基于片段标注的实体关系联合抽取模型（Span-Labeling Based Model， SLM），将实体关系抽取问题转化为片段标注问题，在不过度降低精度的同时，减低了时间空间复杂度，提高了推理速度。

SLM 整体分为编码层和解码层，模型架构图如图 4.1 所示。编码层包含词元编码和片段编码，词元编码使用 ALBERT 预训练模型，片段编码结合预训练模型产生的词元向量、句子向量和片段标记、片段嵌入方法来生成初始片段向量，然后利用 GRU、多头自注意力机制进行片段特征提取；解码层使用片段标签分类器进行关系标签分类，无需实体识别即可单步解码出实体关系三元组。其处理流程在编码层同 SMHS 大同小异，而在解码层 SLM 采用多标签分类的方法，分类出片段标签，而后进行实体关系标签配对，最终解码出实体关系三元组。

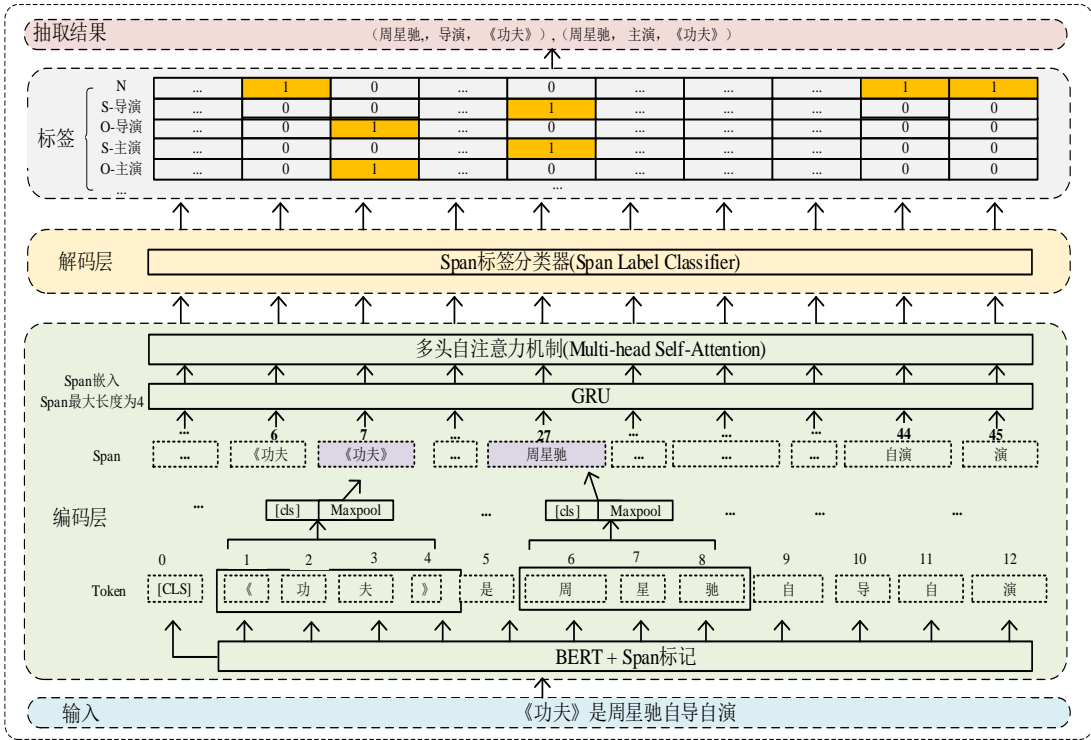


图 4.1 SLM 结构图

第3章所提出的 SMHS 受限于计算时间复杂度和空间复杂度,训练时较为费时,推理时速度较慢而 SLM 仅仅需要进行标签分类,然后标签配对即可抽取实体关系,抽取速度快,模型训练时长短,而精度仅仅是略微下降。

4.2 标签设计

本着简单易懂的原则,SLM 使用的关系标签根据数据集中的关系和主客体进行设计。将标签分为两个部分,标签前缀和标签后缀,使用“-”符号进行分隔。标签前缀用来区分主客实体,使用字母 S 和 O, S (Subject) 表示主实体, O (Object) 表示为客实体;标签后缀用来区分所属关系。例如: S-导演表示所属关系为导演的主体, O-导演表示所属关系为导演的客体,同理 S-演员表示为所属关系为演员的主体, O-演员表示为所属关系为演员的客体。使用标签“N”来表示没有被分配到关系标签,显然每一种关系都有两种标签,故总标签数量为 2 倍关系数量加 1。片段标签示意图如图 4.2 所示。

Span 标签	...	S-导演	N	...	O-导演	...
		S-演员			O-演员	
Span	...	《功夫》	功	...	周星驰	...
输入 语句	《功夫》是周星驰自导自演					

图 4.2 片段标签示意图

标注完后对有相同关系后缀的主客体片段进行配对,若是有多组相同关系后缀的主客体片段则使用“就近原则”进行匹配,进而解码出所有关系三元组,实现单步解码。

4.3 编码层

与 SMHS 相同,同样包括词元编码层和片段编码层。词元编码层通过 BERT 预训练模型的优化版本 ALBERT 获得初步的词元语义向量, ALBERT 在模型参数量上较 BERT 大量减少,能够加快模型训练速度,然后通过不同的片段映射方式进行组合,然后通过 GRU 和多头自注意力机制获得具有深层特征的片段语义向量, GRU 较 LSTM 也能减少模型参数,加快模型训练速度,加快模型推理速度。

4.3.1 词元编码层

选用中文字符级微型 ALBERT 预训练语言模型来对语句进行词元编码，首先根据所选择的中文 ALBERT 模型的词表对文本进行词元化，然后根据设计好的片段标记器和相应的片段映射策略，确定组合后的片段在片段序列中的索引。然后输入词元，提取每个词元的上下文语义信息和句子整体语义信息，关于片段标记和片段映射的相关内容在 2.4 章节已经有详细介绍，关于 ALBERT 的预训练模型相关内容在 2.1.2 章节已经有详细介绍。

假设句子输入表示为 $S = \{s_1, s_2, s_3, \dots, s_n\}$ ，则经过 ALBERT 编码后的向量表示如公式 4.1 所示。

$$H, \bar{H} = ALBERT(S) \quad 4.1$$

其中， $H = \{h_1, h_2, h_3, \dots, h_n\}$ ， $H \in \mathbb{R}^{n \times d}$ ，表示为每个字符被 ALBERT 编码后产生的词元向量， \bar{H} ， $\bar{H} \in \mathbb{R}^d$ ，表示为整个句子被 ALBERT 编码后产生的句子向量。 n 为序列长度， d 为 ALBERT 隐藏状态的维度。

词元编码层较 SMHS 无本质改变，只是为了提升训练速度和推理速度选用了参数数量较小的 ALBERT，提升 SLM 的效率。

4.3.2 片段编码层

SLM 通过将组成片段的每个词元向量进行最大池化后与 ALBERT 的句子向量 CLS 进行拼接的方式构造片段语义向量，能直接提取片段特征，与 SMHS 不同的是 SLM 采用最大池化，来弥补使用较小预训练模型的语义信息不够充分的弱点，最大池化能够最大程度突出词元语义中最显著信息。

假设 $H_{i:j} = \{h_i, h_{i+1}, \dots, h_j\}$ ， $H_{i:j} \in \mathbb{R}^{(j-i+1) \times d}$ ，表示在窗口长度为 w 下，位置索引为 i 的头词元到位置索引为 j 的尾词元中间所有词元所组成的词元向量矩阵，则其所组成的片段语义向量计算公式如公式 4.2—4.5 所示。

$$k = D((i, j)), 0 < i \leq j \leq n, 0 \leq k < m \quad 4.2$$

$$H'_k = \text{Maxpool}(H_{i:j}), H'_k \in \mathbb{R}^d \quad 4.3$$

$$x'_k = \text{Concat}(H'_k, \bar{H}), x'_k \in \mathbb{R}^{2 \times d} \quad 4.4$$

$$x_k = \text{Tanh}(\text{Linear}(x'_k)), x_k \in \mathbb{R}^d \quad 4.5$$

其中 m 为在序列长度为 n ，窗口长度为 w 下所有片段数量，即重新排列后的片段序列的长度， k 为头词元位置索引 i 到尾词元位置索引 j 所组成片段在片段序列中的位置索引， x_k 表示所组成片段的片段语义向量。

对构造出的所有初始片段向量，循环进行式(4)至式(7)的过程，然后根据片段标记器的映射结果，按序组成片段序列。 $X = \{x_1, x_2, x_3, \dots, x_m\}, X \in \mathbb{R}^{m \times d}$ ，表示为所有片段语义向量重新组合成的片段序列。

片段编码层较 SMHS 也无本质改变，只是变换了池化方式，提升重要词元语义特征的语义信息程度。

4.3.3 GRU 和多头自注意力机制层

使用 GRU 和多头自注意力机制加强片段信息交互和深层片段特征提取。

GRU 和多头自注意力的相关内容在 2.2 和 2.3 章节已经由详细介绍

假设 $P = \{p_1, p_2, p_3, \dots, p_m\}$ ，表示为经过 GRU 编码后的片段向量，其计算公式如 4.6 所示。

$$P = \text{GRU}(X), P \in \mathbb{R}^{m \times d} \quad 4.6$$

假设 $T = \{t_1, t_2, t_3, \dots, t_m\}$ ，表示为经过多头注意力机制编码后的片段向量，且使用 z 个平行头来捕获特征，则其中第 i 个头的计算如 4.7 所示， T 的计算公式如 4.8 所示。

$$\text{Head}_i = \text{softmax}\left(\frac{(QW_i^Q)(KW_i^K)}{\sqrt{d/z}}\right)(VW_i^V) \quad 4.7$$

$$T = \text{Concat}(\text{head}_1, \text{head}_2, \dots, \text{head}_z)W^O \quad 4.8$$

其中 $W_i^Q \in \mathbb{R}^{d \times \frac{d}{z}}, W_i^K \in \mathbb{R}^{d \times \frac{d}{z}}, W_i^V \in \mathbb{R}^{d \times \frac{d}{z}}, W^O \in \mathbb{R}^{d \times d}$ ， W^O 是可训练的参数矩阵。

较 SMHS 本层的作用相同仅是稍微对模型进行改变，使用 GRU 来进行片段特征提取，最主要的目的就是减少整个模型的参数量，GRU 已经被证明性能较 LSTM 仅有略微下降，而参数量却显著下降，能有效减少整体模型参数量，加快训练和推理速度。

4.4 解码层

通过对片段标签分类进行解码, 根据分类出的标签的前缀和后缀, 然后进行标签匹配, 进而得出关系三元组。

假设, $I = [i_1, i_2, \dots, i_g]$, I 为标签集合, g 为标签数量。 $\bar{S}(t_i, i_k)$ 表示片段 t_i 被分配 i_k 标签的得分, 公式如 4.9 所示。

$$\bar{S}(t_i, i_k) = Z_k t_i + b_k \quad 4.9$$

其中, $Z_k \in \mathbb{R}^d$, $b_k \in \mathbb{R}^1$ 。

最后计算片段 t_i 被分配 i_k 标签的概率, 公式如 4.10 所示。

$$P_i(\text{Label} = i_k | t_i) = \sigma(\bar{S}(t_i, i_k)) \quad 4.10$$

其中, σ 为 *sigmoid* 激活函数。

解码层是 SLM 较 SMHS 最本质的不同之处, SMHS 所采用的片段多头选择机制, 时间复杂度较高, 训练推理速度慢, 实际使用时消耗大量硬件内存, 而 SLM 采用标签分类时间复杂度降低一个数量级, 仅是在精度上略微下降。

4.5 损失函数

SLM 采用常见的多分类交叉熵损失函数, 公式如 4.11 所示:

$$\text{Loss} = \sum_{t=1}^m \sum_{k=1}^g -\log P_i(\text{Label} = i_k | t_i) \quad 4.11$$

其中, m 为所有片段数量, g 为标签数量。模型训练时同样采用 AdamW 优化器。

4.6 实验与分析

4.6.1 数据集介绍

数据集同样采用中文关系抽取数据集 DuIE2.0, 数据集具体情况, 已在前文介绍, 见章节 3.5.1。

4.6.2 实验环境与参数设置

所采用的实验环境与 SMHS 相同，具体见章节 3.5.2。在实际模型训练中，超参数数值表如表 4.1 所示。

表 4.1 超参数表

超参数名称	超参数值
片段最大长度	15
词元最大长度	62
词元向量维度	128
批次大小	32
学习率	0.00001
最大迭代次数	200
注意力头数	6

4.6.3 实验评价标准

同样使用准确率 Precision、召回率 Recall 和 F1 值来评价实验结果，对于 SLM 也采用稍微宽松的标准：当主实体，客实体，关系类型判断正确就认为是正确的抽取结果，而不是严格的标准：当主实体，主实体类型，客实体，客实体类型，关系类型都判断正确才认为是正确的抽取结果。具体计算公式见章节 3.5.3。

4.6.4 实验结果分析

与 SMHS 相同，为验证模型和辅助模块的有效性和寻找对模型性能有影响的因素，本文设计了三组实验：验证实验、消融实验，根据文本中三元组数量；文本关系重叠类型等因素对模型性能的影响进行实验。

（1）验证实验

所采用的对比基线模型与 SMHS 相同，在加上第三章所提出的 SMHS，为更好地对比模型优劣，将以上模型对句子进行词元编码时都使用相同的 ALBERT 预训练模型，而不是 BERT 预训练模型。实验结果如表 4.2 所示，评价标准中加入了，推理速度指标进行对比。

表 4.2 主要实验结果表

Model	Precision	Recall	F1	推理速度(ms)
ALBERT-tiny+Novel-Tagging	0.701	0.632	0.665	23.1
ALBERT-tiny+CMAN	0.747	0.672	0.708	88.6
ALBERT-tiny+TP-Linker	0.771	0.691	0.729	148.2
ALBERT-tiny+CopyRE	0.751	0.682	0.715	40.1
ALBERT-tiny+WDec	0.767	0.685	0.724	45.2
ALBERT-tiny+SMHS	0.795	0.714	0.752	254.3
ALBERT-tiny+SLM	0.788	0.710	0.747	28.3

从表 4.2 可以看出, SMHS 虽然在精准率, 召回率, F1 值, 均取得了最先进的水平, 但推理速度却比 SLM 高一个数量级, 而性能却只有不到 1 个点的下降。使用 ALBERT-tiny 预训练模型较 BERT 预训练模型, 所有模型性能均有不同程度下降, 是由于为追求推理速度, 所使用的 ALBERT-tiny 仅为 BERT 的参数量的四十分之一, transforms 架构仅为 6 层, 词元向量维度仅为 128, 所表示的文本向量语义信息不如 BERT 丰富。但是, SLM 较其他基线模型也是有不同程度的提高, 是由于 SLM 本质上是基于片段加上精心设计的标签, 从机制上克服了实体嵌套、暴露偏差、关系重叠等问题。

(2) 消融实验

为了进一步探究所设计的方法的影响, 进行了消融实验。实验内容分别为, 去除了多头自注意力机制, 去除 GRU 模块。实验结果如表 4.3 所示。

表 4.3 消融实验结果表

研究内容	Precision	Recall	F1
去除 Multi Head Self-Attention	0.765	0.700	0.731
去除 GRU	0.751	0.689	0.718
正常模型	0.788	0.710	0.747

从表 4.3 可以看出, 去除了多头自注意力机制导致模型的 F1 降低了 1.6%, 说明多头注意力机制的片段深层特征提取对模型是有效的, 是由于仅使用 GRU 进行片段

编码仅能加强片段之间的交互及依赖关系，无法对深层片段特征信息进行提取；去除了 GRU 模块导致模型的 F1 降低 2.9%，是由于去除了 GRU 模块，缺少了片段间的交互信息和片段依赖关系。消融实验证明，模型中为提升关系抽取效果所设计的辅助模块是有效的。

(3) 影响因素实验

为了更加清晰地探究，相关因素对模型性能影响，设计了多组影响因素实验，具体有：文本中三元组数量对模型的影响程度；文本关系重叠的类型对模型的影响程度；不同片段映射策略对模型的影响；不同的窗口长度对模型的影响。实验结果如表 4.4，4.5，4.6，4.7 所示。

表 4.4 三元组数量实验结果表

三元组数量 (N)	Precision	Recall	F1
N=1	0.837	0.755	0.793
N=2	0.791	0.725	0.756
N=3	0.751	0.711	0.730
N=4	0.721	0.681	0.700
N>=5	0.688	0.669	0.678

表 4.5 关系重叠实验结果表

关系重叠类型	Precision	Recall	F1
Normal	0.791	0.722	0.755
SEO	0.779	0.702	0.738
EPO	0.768	0.704	0.731

表 4.6 片段映射策略实验结果表

片段映射策略	Precision	Recall	F1
Start	0.788	0.710	0.747
End	0.775	0.702	0.736
Length	0.741	0.671	0.704

表 4.7 窗口长度实验结果表

窗口长度	Precision	Recall	F1
5	0.662	0.602	0.630
6	0.677	0.608	0.640
7	0.691	0.609	0.647
8	0.711	0.641	0.674
9	0.712	0.655	0.682
10	0.719	0.661	0.688
11	0.726	0.685	0.704
12	0.753	0.688	0.719
13	0.765	0.701	0.731
14	0.779	0.715	0.745
15	0.788	0.710	0.747

从表 4.4 中可以看出,当文本中三元组数量较少时,SLM 的性能甚至超出了 SMHS,但随着文本三元组的数量的上升,SLM 的性能下降幅度略大,但仍旧保持了不错的水平。这是由于 SLM 抽取实体关系三元组才用的是标签匹配的方式,当三元组数量多时,“就近原则”的匹配方式,不免会带来匹配错误,导致模型性能下降,这也说明在应对复杂文本时,SLM 性能不如 SMHS,鲁棒性不如 SMHS,比较适用单一领域的简单文本实体关系三元组抽取。

从表 4.5 中可以看出,面对不同关系重叠类型时,SLM 性能仅是略微下降,这是由于,采用了多标签分类的方案进行标签分类可以进行多轮标注,不会遗漏该要抽取的实体关系标签,能较好克服关系重叠问题。

从表 4.6 中可以看出,SLM 和 SMHS 相同,都是采用 start 映射策略时,取得最优效果,更进一步说明了,在进行片段级别中文实体关系抽取时,头尾词元所含语义信息的重要性,特别是头词元。

从表 4.7 中可以看出,于 SMHS 同样相同,随着窗口长度上升,SLM 的性能会有逐步的上升,但 SLM 并未出现下降趋势,是由于 SLM 的解码层方式较为简单,即使片段序列长度过长,对标签分类所造成的影响也很小,反而是所包含更多的词元信息能够有效帮助模型提高性能。

4.7 本章小结

本章主要是全面地介绍所提出的基于片段标注的实体关系联合抽取模型 **SLM** 的相关内容，包括模型架构、模型处理步骤、计算公式、数据介绍、实验分析等。实验结果表明，所提出的 **SLM** 比目前主流的模型效果都要好，较第三章所提出的 **SMHS** 性能略微下降，但模型在训练速度、推理速度方面有较大提升，面对复杂语境，三元组数量较多时，仍可以保持应有的性能，比 **SMHS** 在实际应用方面略胜一筹，同时实验结果说明 **SLM** 能有效解决实体嵌套、暴露偏差、关系重叠问题，也验证了 **SLM** 和所提出的辅助模块的有效性。

第 5 章 总结与展望

5.1 总结与结论

实体关系抽取，作为构建知识图谱的基本上游任务，实体关系抽取性能的好坏直接关系到知识图谱构建的好坏。当前，随着深度学习、计算机硬件和语言预训练模型不断发展，实体关系抽取模型不断推陈出新，模型性能节节攀升，但仍有几个痛点问题亟待解决如实体嵌套，关系重叠，暴露偏差等问题。这些问题一直以来对实体关系抽取模型的性能影响较大，且难以处理。

针对上述问题，通过对以往研究人员的实体关系抽取模型的研究和扩展，提出了一种片段标记、三种片段映射策略，以及两个基于片段处理的实体关系抽取模型，分别为：基于片段多头选择的实体关系联合抽取模型 SMHS (Span based Multi Head Selection) 和基于片段标注的实体关系联合抽取模型 SLM (Span-Labeling Based Model)。

基于片段的模型，通过枚举词元排列的方法从机制上根本地解决了实体嵌套问题，三种不同的映射策略来将枚举出的词元排列重新平铺转化为片段排列，分别为：start 策略，通过在固定的窗口长度内，将共享头词元的片段进行收集排列；end 策略，通过在固定的窗口长度内，将共享尾词元的片段进行收集排列；length 策略，通过在固定的窗口长度内，将相同长度的片段进行收集排列，实验结果表明，start 策略更加适合中文文本，这和中文文本的特点是相契合的。

SMHS 使用 BERT 预训练模型作为词元向量表示，通过片段标记以及片段嵌入的方法构造片段向量表示，使用 LSTM 和多头自注意力机制进行片段深层特征提取，最后使用多头选择机制实现单步的实体关系抽取，且引入了片段分类来进行辅助训练，为关系抽取任务提供了良好的片段类型信息和片段类型约束，相比较于各种基线模型，在中文关系抽取数据集 DuIE2.0 上，F1 值取得了最佳结果，验证了模型的有效性。

由于 SMHS，训练时所需的硬件资源较多，推理的时间复杂度较高，故提出，空间复杂度较低，推理速度较快的 SLM。SLM 使用 ALBERT-tiny 预训练模型作为词元向量表示，同样通过片段标记和片段嵌入的方法构造片段向量表示，使用 GRU 和多头自注意力机制进行片段深层特征提取，最后使用多层标签分类方法，识别出片段实体关系标签，从而实现单步实体关系抽取，相比较于各种基线模型，在中文关系抽取数据集 DuIE2.0 上，F1 值取得了较佳结果，验证了模型的有效性。

SMHS 和 SLM 相比较而言，SLM 胜在所需的训练的硬件资源较小，训练速度和推理速度比 SMHS 快一个数量级，而性能仅有小幅度下降，但面对复杂语境时，处

理能力不如 SMHS，更适合大范围的单一领域的简单文本的实体关系抽取，实际应用价值高；SMHS 胜在模型精度高，性能强，面对任何复杂语境均能较好地抽取实体关系三元组，不存在性能瓶颈，但由于训练和推理速度较一般模型而言较慢，在实际应用中时，需要在窗口长度和推理速度之间进行取舍，以保证模型的应用价值。

5.2 展望

本文所提出的 SMHS 和 SLM 虽然取得了不错效果，但仍有一些需要改进和提升的地方，以及进一步探究的余地。

对于 SMHS 来说，虽然能有效解决关系重叠，误差积累，暴露偏差等问题，但仍有不足之处，例如，在进行构造片段语义向量时需耗费大量内存空间、进行多头关系选择时计算复杂度高，导致无法对过长的文本进行建模，在构造片段向量表示做法略显粗糙，引入片段类型信息和片段类型约束的方式并不直接。如何降低计算复杂度，更精细高效地构造片段向量，更加直接、显式地引入片段类型信息和片段类型约束引入是进一步研究的重点。

对于 SLM 来说，同样能有效解决和缓解关系重叠，误差积累，暴露偏差等问题，但还有探索的余地，例如：文本内有多条具有相同关系的关系三元组时，采用就近原则来进行关系主客体的匹配，处理手法过于简单。如何更好地匹配同一关系主客体和解决复杂语境下抽取性能不足的问题也是后续的研究方向和研究重点。

除此之外，对于最大窗口长度的选取及片段序列长度和片段语义信息的丰富程度两者之间的内在联系也还需要探究实验；所提出的三种片段映射策略的实际应用区别还不是很明了，将进一步研究；当前的实验是基于中文文本，或可将模型推广到英文甚至其他语言环境中，下一步也将进行实验；由于开源中文实体关系抽取数据集的欠缺，没有验证模型在更多关系类别的数据中的表现，将进一步寻找或构造合适的中文实体关系数据集进行验证。

综上所述，在未来的研究中，主要从模型处理方式的改良和模型模块内在联系的探究为重点，将模型进行推广和构造额外的中文实体关系抽取数据集也是需要研究的方向。

参考文献

- [1] 武文雅,陈钰枫,徐金安,张玉洁.中文实体关系抽取研究综述[J].计算机与现代化,2018(08):21-27+34.
- [2] 谢德鹏,常青.关系抽取综述[J].计算机应用研究,2020,37(07):1921-1924+1930.
- [3] Chinchor N, Marsh E. Muc-7 information extraction task definition [C]//Proceedings of the 7th Message Understanding Conference(MUC-7).1998:359-367.
- [4] Hendrickx I, Kim S N, Kozareva Z, et al. Semeval-2010 task8: Multi-way classification of semantic relation between pairs of nominal [C]//Proc of Workshop on Semantic Evaluations: Recent Achievements and Future Directions. 2009:94-99.
- [5] 李冬梅,张扬,李东远,林丹琼.实体关系抽取方法研究综述[J].计算机研究与发展,2020,57(07):1424-1448.
- [6] 邓攀,樊孝忠,杨立公用语义模式提取实体关系的方法[J].计算机工程,2007,33(10):212-214.
- [7] 温春,石昭祥,辛元.基于扩展关联规则的中文非分类关系抽取[J].计算机工程,2009,35(24):63-65.
- [8] 王传栋,徐娇,张永.实体关系抽取综述[J].计算机工程与应用,2020,56(12):25-36.
- [9] 鄂海红,张文静,肖思琪,程瑞,胡莺夕,周筱松,牛佩晴.深度学习实体关系抽取研究综述[J].软件学报,2019,30(06):1793-1818.
- [10] Kambhatla N. Combining lexical, syntactic, and semantic features with maximum entropy models for extracting relations [C]//Proceedings of ACL on Interactive Poster and Demonstration Sessions, 2004: 22-26.
- [11] Zhou G D, Sun J, Zhang J, et al. Exploring Various Knowledge in Relation Extraction [C]//Proceedings of the 43rd Annual Meeting on Association for Computational Linguistics, 2005:427 -434.
- [12] Jiang J, Zhai C X. A Systematic exploration of the feature space for relation extraction [C]//Proceedings of the Human Language Technology Conference of the North American Chapter of the Association for Computational Linguistics, 2007: 113-120.
- [13] 车万翔,刘挺,李生.实体关系自动抽取[J].中文信息学报,2005(02):1-6.
- [14] 郭喜跃,何婷婷,胡小华,陈前军.基于句法语义特征的中文实体关系抽取[J].中文信息学报,2014,28(06):183-189.

- [15] 高俊平,张晖,赵旭剑,杨春明,李波.面向维基百科的领域知识演化关系抽取[J].计算机学报,2016,39(10):2088-2101.
- [16] 甘丽新. 基于句法和语义分析的中文实体关系抽取[D].江西财经大学,2017.
- [17] 张晓峰. 基于核方法的实体关系抽取研究[D].东南大学,2016.
- [18] Zelenko D, Aone C, Richardella A. Kernel methods for relation extractio[J].The Journal of Machine Learning Research,2003,3:1083-1106.
- [19] Culotta A, Sorensen J. Dependency tree kernels for relation extraction[C]//The 42nd Meeting of Association for Computational Linguistics, 2004.
- [20] Zhang X F, Gao Z Q, Zhu Man. Kernel methods and its application in relation extraction[C]//Proc of the Int Conf on Computer Science and Service System (CSSS).Piscataway, NJ: IEEE,2011: 1362-1365.
- [21] 刘克彬,李芳,刘磊,韩颖.基于核函数中文关系自动抽取系统的实现[J].计算机研究与发展,2007(08):1406-1411.
- [22] 虞欢欢,钱龙华,周国栋,朱巧明.基于合一句法和实体语义树的中文语义关系抽取[J].中文信息学报,2010,24(05):17-23.
- [23] 陈鹏. 基于多核融合的中文领域实体关系抽取研究[D].昆明理工大学,2014.
- [24] Socher R, Huval B, Manning CD, Ng AY. Semantic compositionality through recursive matrix-vector spaces.In: Proc. of the Joint Conf. on Empirical Methods in Natural Language Processing and Computational Natural Language Learning.2012.1201-1211.
- [25] Zeng D J, Liu K, Lai S W, et al.Relation classification via convolutional deep neural network [C]//Proc of the 25th Int Conf on Computational Linguistics. Stroudsburg: ACL,2014: 2335-2344.
- [26] Liu CY ,Sun WB, Chao WH,et al.Convolution neural network for relation extraction[C]//Proc of the Int Conf on Advanced Data Mining and Applications.Berlin:Springer,2013: 231-242.
- [27] Nguyen T H, Grishman R. Relation extraction: Perspective from convolutional neural networks [C]//Proc of the 1st Workshop on Vector Space Modeling for Natural Language Processing.Stroudsburg:ACL,2015:39-48.
- [28] 孙建东,顾秀森,李彦,徐蔚然.基于COAE2016 数据集的中文实体关系抽取算法研究[J].山东大学学报(理学版),2017,52(09):7-12+18.
- [29] 高丹,彭敦陆,刘丛.海量法律文书中基于CNN的实体关系抽取技术[J].小型微型计算机系统,2018,39(05):1021-1026.

- [30] 吴天昊, 古丽拉·阿东别克. 基于神经元块级别注意力机制的LSTM关系抽取[J]. 计算机应用研究, 2020, 37(S2): 76-79.
- [31] Xu K, Feng Y, Huang S, et al. Semantic relation classification via convolutional neural networks with simple negative sampling[J]. arXiv preprint arXiv:1506.07650, 2015.
- [32] Zhang S, Zheng D, Hu X, et al. Bidirectional long short-term memory networks for relation classification[C] //Proceedings of the 29th Pacific Asia conference on language, information and computation. 2015: 73-78.
- [33] Zhong Z, Chen D. A frustratingly easy approach for joint entity and relation extraction. 466[J]. arXiv preprint arXiv:2010.12812, 2020, 467.
- [34] Gupta P, Schhitze H, Andrassy B. Table filling multi-task recurrent neural network for joint entity and relation extraction[C]//Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers. 2016: 2537-2547.
- [35] Zheng S, Wang F, Bao H, et al. Joint Extraction of Entities and Relations Based on a Novel Tagging Scheme[C]//Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers). 2017: 1227-1236.
- [36] Dai D, Xiao X, Lu Y, et al. Joint Extraction of Entities and Overlapping Relations Using Position-Attentive Sequence Labeling[J]. Proceedings of the AAAI Conference on Artificial Intelligence, 2019, 33:6300-6308.
- [37] Zeng D, Zhang H, Liu Q. CopyMTL: Copy Mechanism for Joint Extraction of Entities and Relations with Multi-Task Learning[J]. Proceedings of the AAAI Conference on Artificial Intelligence, 2020, 34(5):9507-9514.
- [38] Zhang R H, Liu Q, Fan A X, et al. Minimize Exposure Bias of Seq2Seq Models in Joint Entity and Relation Extraction[C]//Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: Findings. 2020: 236-246.
- [39] Wei Z, Su J, Wang Y, et al. A Novel Cascade Binary Tagging Framework for Relational Triple Extraction[C]//Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics. 2020: 1476-1488.
- [40] Miwa M, Bansal M. End-to-End Relation Extraction using LSTMs on Sequences and Tree Structures[C]//Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers). 2016: 1105-1116.

-
- [41] Bekoulis G, Deleu J, Demeester T, et al. Joint entity recognition and relation extraction as a multi-head selection problem[J]. *Expert Systems with Applications*, 2018, 114: 34-45.
 - [42] Katiyar A, Cardie C. Going out on a limb: Joint extraction of entity mentions and relations without dependency trees[C]//*Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. 2017: 917-928.
 - [43] Li X, Yins F, Suns Z, et al. Entity-Relation Extraction as Multi-Turn Question Answering[C]//*Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*. 2019: 1340-1350.
 - [44] Dixit K, Al-onazans Y. Span-level model for relation extraction[C]//*Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*. 2019: 5308-5314.
 - [45] Fu T J, Li P H, Ma W Y. GraphRel: Modeling text as relational graphs for joint entity and relation extraction[C]//*Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*. 2019: 1409-1418.
 - [46] Wang Y, Yu B, Zhang Y, et al. TPLinker: Single-stage Joint Extraction of Entities and Relations Through Token Pair Linking[C]//*Proceedings of the 28th International Conference on Computational Linguistics*. 2020: 1572-1582.
 - [47] Sui D, Chen Y, Liu K, et al. Joint entity and relation extraction with set prediction networks[J]. *arXiv preprint arXiv:2011.01675*, 2020.
 - [48] Mikolov T, Chen K, Corrado G, et al. Efficient estimation of word representations in vector space[J]. *arXiv preprint arXiv:1301.3781*, 2013.
 - [49] Bengio Y, Ducharme R, Vincent P, et al. A Neural Probabilistic Language Model[J]. *Journal of Machine Learning Research*, 2003, 3: 1137-1155.
 - [50] Peters M, Neumann M, Iyyer M, et al. Deep Contextualized Word Representations[J]. *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, 2018.
 - [51] Devlin J, Chang M W, Lee K, et al. Bert: Pre-training of deep bidirectional transformers for language understanding[J]. *arXiv preprint arXiv:1810.04805*, 2018.

-
- [52] Cho K, Van Merriënboer B, Gulcehre C, et al. Learning phrase representations using RNN encoder-decoder for statistical machine translation[J]. arXiv preprint arXiv:1406.1078, 2014.
 - [53] Schmidhuber J. Deep learning in neural networks: An overview[J]. Neural networks, 2015, 61: 85-117.
 - [54] Hochreiter S, Schmidhuber J. Long short-term memory[J]. Neural computation, 1997, 9(8): 1735-1780.
 - [55] Cho K, Van Merriënboer B, Gulcehre C, et al. Learning phrase representations using RNN encoder-decoder for statistical machine translation[J]. arXiv preprint arXiv:1406.1078, 2014.
 - [56] Vaswani A, Shazeer N, Parmar N, et al. Attention is all you need[J]. Advances in neural information processing systems, 2017, 30.
 - [57] Peng H, Gao T, Han X, et al. Learning from context or names? an empirical study on neural relation extraction[J]. arXiv preprint arXiv:2010.01923, 2020.
 - [58] Caruana R. Multitask learning[J]. Machine learning, 1997, 28(1): 41-75.
 - [59] Li S, He W, Shi Y, et al. Duie: A large-scale chinese dataset for information extraction[C]//CCF International Conference on Natural Language Processing and Chinese Computing. Springer, Cham, 2019: 791-800.
 - [60] Kingma D P, Ba J. Adam: A method for stochastic optimization[J]. arXiv preprint arXiv:1412.6980, 2014.
 - [61] Zhao S, Hu M, Cai Z, et al. Modeling dense cross-modal interactions for joint entity-relation extraction[C]//Proceedings of the Twenty-Ninth International Conference on International Joint Conferences on Artificial Intelligence. 2021: 4032-4038.
 - [62] Zeng X, Zeng D, He S, et al. Extracting relational facts by an end-to-end neural model with copy mechanism[C]//Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers). 2018: 506-514.
 - [63] Nayak T, Ng H T. Effective modeling of encoder-decoder architecture for joint entity and relation extraction[C]//Proceedings of the AAAI conference on artificial intelligence. 2020, 34(05): 8528-8535.
 - [64] Lan Z, Chen M, Goodman S, et al. Albert: A lite bert for self-supervised learning of language representations[J]. arXiv preprint arXiv:1909.11942, 2019.

致 谢

行文至此，感慨万千。

时光如流，岁月匆匆。三年求学生涯虽多有波折，幸有师长激励，同窗互勉，父母支持，心中感激之情，无以言表，仅以三言两语聊表谢意。

爱国敬业，求实创新。感谢母校——长春工业大学，提供了良好的生活、学习环境，先进、开阔的实验平台，让我能顺利开展论文实验，领悟工大精神。

吉祥宝地，多彩吉林。感谢美丽的吉林，在吉林省，求学七载，让我亲眼领略到北国风光；让我真切感受到东北人民正直、豪爽的人格魅力；让我深刻体悟到积极向上，努力奋斗的人生意义。

桃李不言，下自成蹊。感谢导师赵辉教授，导师才德兼备，让我高山仰止，从生活到学业方面都给予我无微不至的关心和认真严谨的指导，使我受益颇多。感谢 204 实验室所有的指导老师，在我困惑、迷茫时给予的帮助。

一朝相知，终生知己。感谢 204 实验室的所有同学，彭雪，秦硕，许文军，朱海东，黄思佳，杨鑫；感谢我的室友，王浩然，张指旗，李浩，张灏铭，杨承林，刘伟业，孟凡磊。感谢他们，在学习和生活上给予我的帮助和包容。

春晖寸草，山高海深。感谢我伟大的父母，默默关心我，在我身后给予物质、精神上的支持和无私的爱，原谅我的不成熟和过错，无条件地信任我。

道阻且长，行则将至。感谢认真努力、坚持不懈的自己。

行而不辍，未来可期。祝愿母校，蒸蒸日上，更创辉煌；祝愿吉林，愈加美丽，愈加富强；祝愿导师，桃李芬芳，万事如意；祝愿同窗，一帆风顺，各有所成；祝愿父母，身体康健，万事顺遂。

路漫漫其修远兮，吾将上下而求索！在未来的道路上，我会更加刻苦努力，不负母校栽培，不负师恩，不负父母期望！

作者简介

郑肇谦，男，1996 年 11 月出生，汉族

工作单位：长春工业大学

攻读硕士学位期间研究成果

一、发表论文：

- [1] 郑肇谦,韩东辰,赵辉.单步片段标注的实体关系联合抽取模型[J/OL].计算机工程与应:1-11[2022-06-02].（已录用，网络首发，预计见刊时间 2023 年 4 月）
- [2] 彭雪,赵辉,郑肇谦,庞海婷.融合多种嵌入表示的中文命名实体识别[J].长春工业大学学报, 2022,43(01).

二、授权发明专利：

- [1] 基于自注意力机制和卷积神经网络的文本分类算法[P].ZL202110582336.3，2021.9.13,3/3.