

摘要

基于深度学习指针标注的联合关系抽取研究

实体关系抽取任务的目标是从非结构化文本中预测实体及其相互间的关系，生成结构化数据，是构建知识库的核心技术之一。面对复杂语境下的重叠关系问题，当前主流研究主要采用两步式的联合关系抽取方法。尽管这些方法取得一定程度的成功，但仍面临曝光误差和头尾实体交互不足的问题，影响关系抽取性能。面对以上挑战，本文提出相应的方法并取得显著的效果，主要贡献包括：

首先，本文提出一种基于全局特征的指针标注联合关系抽取模型。该模型采用指针标注框架，能有效地解决复杂语境中的重叠关系问题。同时，通过使用 BERT 对句子进行编码，结合多头注意力机制和多核卷积神经网络提取多粒度的全局特征，用于指导头实体的预测，有助于提高头实体识别的准确性，降低曝光误差。

其次，在基于上述关系抽取模型的基础上，本文设计一种自适应关系-实体信息融合模块，提出一种融合关系与实体信息的联合关系抽取模型。自适应关系-实体信息融合模块由一个关系注意力模块和两个门控融合模块组成。其中，关系注意力模块主要负责建立实体与关系间的交互，以获取不同关系对每个词的重要程度，两个门控融合模块主要负责将关系嵌入信息和头实体信息与每个词嵌入进行融合，用于指导关系和尾实体的预测，加强头尾实体任务间的交互。

最后，为验证本文提出方法的有效性，本文在 NYT 和 WebNLG 数据集上进行基线模型对比实验、消融实验以及重叠问题的评估实验。实验结果显示，本文最终模型的 F1 值在 NYT 数据集上达到 91.6%，在 WebNLG 数据集上达到 92.6%，验证了本文方法在解决重叠关系问题上具有优秀的能力。

关键词：

曝光误差，联合关系抽取，重叠关系三元组，融合关系表达特征

Abstract

Research on Joint Entity Relation Extraction Based on Deep Learning with Pointer Annotation

The goal of entity relation extraction task is to predict entities and their relationships from unstructured text, and then convert it into structured data, which is one of the core technologies for building knowledge bases. Facing the overlapping relationship problem in complex contexts, the current mainstream research mainly adopts a two-step joint relation extraction method. Although these methods have achieved a certain degree of success, they still face the issues of exposure error and insufficient head-tail entity interaction, which affect the performance of relation extraction. In response to these challenges, this paper propose corresponding methods and achieves significant results. The main contributions include:

Firstly, this paper propose a pointer-labeling joint relation extraction model based on global features. The model employs a pointer-labeling framework, which effectively tackles overlapping relationship issues in complex contexts. Simultaneously, by encoding sentences with BERT, and combining multi-head attention mechanism and multi-kernel convolutional neural networks to extract multi-granular global features for guiding the prediction of head entities, it helps to improve the accuracy of head entity recognition and reduce exposure errors.

Secondly, building upon the aforementioned relation extraction model, this paper designs an adaptive relation-entity information fusion module and proposes a joint relation extraction model that integrates both relation and entity information. The adaptive relation-entity information fusion module consists of a relation attention module and two gated fusion modules. The relation attention module is responsible for establishing the interaction between entities and relations, obtaining the importance of different relationships for each word, while the two gated fusion modules combine relation embeddings and head entity information with each word embedding, guiding

the prediction of relations and tail entities, and strengthening the interaction between the head and tail entity tasks.

Lastly, in order to verify the effectiveness of the proposed methods, this paper conducts baseline model comparison experiments, ablation experiments, and evaluation experiments for overlapping issues on the classic NYT and WebNLG datasets. The experimental results show that the final model proposed in this paper achieves an F1 score of 91.6% on the NYT dataset and 92.6% on the WebNLG dataset. demonstrating These results demonstrate that the model proposed in this paper has a significant advantage in addressing relationship overlapping issues.

Keywords:

Exposure Error, Joint Relation Extraction, Overlapping Relation triplets, Fused Relation Expression Features

关于学位论文使用授权的声明

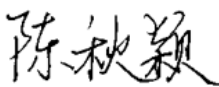

本人完全了解吉林大学有关保留、使用学位论文的规定，同意吉林大学保留或向国家有关部门或机构送交论文的复印件和电子版，允许论文被查阅和借阅；本人授权吉林大学可以将本学位论文的全部或部分内容编入有关数据库进行检索，可以采用影印、缩印或其他复制手段保存论文和汇编本学位论文。

（保密论文在解密后应遵守此规定）

论文级别： 硕士 博士

学科专业： 计算机技术

论文题目： 基于深度学习指针标注的联合关系抽取研究

作者签名：  指导教师签名： 

2023年5月22日

目 录

第 1 章 绪论.....	1
1.1 研究背景及意义	1
1.2 课题的研究现状	2
1.2.1 基于流水线关系抽取方法	2
1.2.2 实体关系联合抽取方法	4
1.3 本文研究内容	6
1.4 本文组织结构	7
第 2 章 相关技术介绍	8
2.1 全连接神经网络	8
2.2 卷积神经网络	9
2.3 注意力机制	11
2.4 预训练语言模型 BERT	13
2.5 本章小结	14
第 3 章 基于全局特征的指针标注联合关系抽取	15
3.1 问题定义	15
3.2 JREPA_GF 模型	17
3.2.1 句子编码层	19
3.2.2 基于 BERT 的全局特征处理层	20
3.2.3 头实体识别层	23
3.2.4 关系-尾实体识别层	24

3.3 本章小结	26
第 4 章 基于关系增强的指针标注联合关系抽取	27
4.1 JREPA_RE 模型	27
4.1.1 自适应关系-实体融合模块	28
4.1.2 关系-尾实体标注器	30
4.2 模型的训练流程及实现	31
4.2.1 JREPA_RE 模型的训练流程	31
4.2.2 JREPA_RE 模型的伪代码实现	33
4.2.3 JREPA_RE 模型优化方法	34
4.3 本章小结	34
第 5 章 实验设计与分析	35
5.1 实验数据及实验评价指标	35
5.2 实验环境和超参数设置	37
5.2.1 实验环境	37
5.2.2 实验超参数设置	38
5.3 对比实验及结果分析	38
5.3.1 基线模型及结果分析	38
5.3.2 消融实验的对比及分析	41
5.3.3 重叠问题的实验评估	43
5.4 本章小结	46
第 6 章 总结与展望	47
6.1 工作总结	47

6.2 工作展望	48
参考文献.....	49
作者简介及在学期间所取得的科研成果	55
致谢.....	56

第1章 绪论

1.1 研究背景及意义

近年来，随着人工智能相关技术的高速发展，自然语言处理研究开始扮演越来越重要的角色，逐渐成为人工智能进一步发展的关键支撑技术。面对日益激增、种类繁多、形式丰富且非结构化的互联网数据，将其转变为易于理解和利用的结构化信息，实现数据资源的最大价值化，已经成为自然语言处理研究需要解决的关键性问题，信息抽取技术^[1]为实现此目标而诞生。信息抽取技术主要任务为从文本中自动、高效提取出特定的实体、关系和事件等类型信息，形成结构化数据。其中，实体关系抽取是信息抽取技术的子领域之一，是自然语言处理领域中的一项重要研究内容，是自动构建知识图谱（例如：Knowledge Vault^[2]、YAGO^[3]、DBpedia^[4]以及 FreeBase^[5]等）的关键手段，且其在互联网语义搜索、知识融合、智能问答与大数据分析等任务中发挥出巨大作用，具有重要的应用价值。实体关系抽取的基本任务是从纯文本中识别出句子中的实体^[6]，并判断多个实体组成的实体对之间的关系类别。实体对和关系的组成形式可表示为（“头实体”，“关系”，“尾实体”）。实体关系抽取可以拆分为两个任务：实体识别任务^[7]和关系分类任务^[8]。根据这两个任务是否独立，将关系抽取任务分为基于流水线关系抽取和联合关系抽取。

目前，已有许多高效的工作用于处理简单语境下的关系抽取。然而，在面对复杂语境中的重叠关系问题，例如：在一个句子中包含多个三元组，且这些三元组中的实体具有多个关系，大多数现有方法的处理效果并不理想。其中，基于流水线关系抽取主要先识别出所有实体，再预测实体之间的关系，但这忽视实体识别和关系分类任务间的语义依赖关系，可能导致错误传播和实体冗余，从而影响关系分类的性能。联合关系抽取方法通常采用参数共享的手段来统一实体识别和关系抽的模型架构，虽然能有效地缓解流水线方法出现的缺点，但由于未能充分利用实体和关系之间的语义信息，容易产生曝光误差问题、实体模型和关系模型的语义交互不足问题。因此，充分挖掘句子中的潜在语义信息

及实体与关系间的有效信息，加强实体抽取任务和关系抽取任务的交互，对于提高模型在复杂语境下的关系抽取性能具有重要意义。

1.2 课题的研究现状

关系抽取的研究方向主要分为两类：基于流水线的关系抽取方法和实体关系联合抽取方法。其中，基于流水线的关系抽取方法自问世以来就备受关注，而实体关系联合抽取方法则是针对基于流水线关系抽取方法存在的问题而提出。相较于流水线关系抽取方法，联合关系抽取方法更适合解决复杂语境下的关系抽取问题。因此，联合关系抽取的研究被逐渐被广泛学者关注。本文将详细介绍这两种方法。

1.2.1 基于流水线关系抽取方法

基于流水线关系抽取方法将实体关系抽取任务分解为实体识别任务和关系抽取任务两个独立的子任务。具体而言，先训练一个实体识别模型来抽取句子中的全部潜在的实体，再训练一个关系分类模型，将抽取到的实体作为输入并进行关系分类。因此，基于流水线关系抽取方法依赖实体对的提取。根据不同类型的特征工程，可将其划分为基于特征向量的关系抽取方法、基于核函数的关系抽取方法和基于神经网络的关系抽取方法。

(1) 基于特征向量的关系抽取方法

基于特征向量的关系抽取方法的主要工作是根据文本信息提取出合适的特征，并构建适用于任务的分类器。Kambhatla^[9]等人采用最大熵分类模型，根据提取的实体上下文信息，构造依存关系树及句法分析树等特征进行关系分类，结果表明，实体潜在的语义特征能够增强关系的语义表示。在此基础上，Zhou^[10]等人采用句子的词块信息，有效地捕捉文本的句法信息，并采用 WordNet 和 Name List 来增强文本的语义信息，进一步提升关系分类效果。Wang^[11]等人则在关系分类中引入单词的词性、实体的层级关系和句子的简化逻辑形式等特征。为达到与人工表现级别相当的抽取效果，Culotta^[12]等人将文本信息融合，利用词性标注和同义词网络对依赖树的结点进行特征增强，并采用

支持向量机进行分类。由于这类方法需要依赖专家知识来设计特征，且有效的句法特征、词汇特征和语义特征的种类相对较少。因此，基于核函数的关系抽取方法被提出。

(2) 基于核函数的关系抽取方法

基于核函数的关系抽取方法最早由 Zelenko^[13]等人提出，其核心思想是在句子的结构树、子序列上构建核函数，利用核函数计算不同关系实例的特定表示间的相似度，用于关系分类，能够有效解决特征数量不足问题。Collins^[14]等人提出采用卷积解析树核函数计算两个句法树的相似性，其本质思想是计算句法树中结构相同的子树数量。在此基础上，Khayyamian^[15]等人提出语法树核函数，进一步提高基于核函数来关系抽取任务的性能。鉴于依存树中实体对和关系的最短依存路径能较好地表示目标关系的特性，Bunescu^[16]等人提出基于依存路径的核函数，以计算最短依存路径的相似性。虽然利用核函数可以通过隐式地在高维空间中计算句子的相似度来提高模型的分类性能，但这类方法进行句法分析时严重依赖自然语言处理工具，在大规模语料中的关系抽取任务效果不佳，且不能保证句法分析的正确性，从而影响关系抽取的效果。

(3) 基于神经网络的流水线方法

鉴于神经网络（卷积神经网络^[17]、循环神经网络^[18]、长短期记忆网络以及卷积神经网络^[19]等）能自动地提取特征，避免了传统手工设计特征产生的错误逐级传递问题，研究者们逐渐将其广泛应用于基于流水线的关系抽取任务中，极大地提升了关系抽取任务的性能和效率。

为了研究如何对包含关系和实体信息的句子进行编码以及如何输出一个定长的关系表示，Soares^[20]人采用 BERT，根据不同的输入、输出策略来测试关系抽取性能，为使用 BERT 进行关系抽取打下基础。为提取一个句子中多个实体对和关系，Wang^[21]等人提出采用预训练模型 BERT 和池化机制得到所有实体，再通过向线性层融入实体和单词间的位置信息来增强实体对的关系分类，这成为使用 BERT 进行关系抽取的常用做法。Patrick Verga^[22]等人将句子级别关系抽取推广到文档级别的关系抽取，将关系分类由通常的多分类问题转为二分类问题，并通过引入实体识别任务来增强多任务学习，实现多关系实体对的提取。2020年，Zhong^[23]等人采用片段排列的标注方式解决实体识别模型中实体嵌套问

题，并通过显性标记实体片段的方式来提升模型性能。2022年，Ye^[24]等人在Zhong的基础上考虑实体片段间的相互关系，在实体识别任务中设计一种面向邻居片段的打包策略来建模实体边界信息，在关系抽分类任务中打包每个头实体以及可能的尾实体，通过建模相同头实体的跨度信息来进行信息交互。在ACE04和ACE05数据集上F1值超越先前的模型，F1值分别为71.1%和66.5%。

目前，在多个领域中，基于流水线的关系抽取方法已经达到了较高的抽取性能。然而，基于流水线的关系抽取方法存在一些不足之处：实体识别任务的性能直接决定关系分类任务的性能，若实体识别任务出现错误，则在关系分类任务中就没有机会进行更正，容易导致错差传播。此外，将实体识别任务与关系抽取任务单独处理，忽视这两个子任务的潜在联系。

1.2.2 实体关系联合抽取方法

为了解决基于流水线方法存在的弊端，研究者提出了一种联合实体和关系的联合关系抽取方法。然而，在复杂语境下，重叠关系问题是实体关系联合抽取任务的难点。因为一个句子中的相同的实体可能会存在多种关系类型。例如句子：“华盛顿是美国的首都”，“华盛顿”和“美国”这两个实体间存在“首都”和“包含”两类关系类型。目前，联合关系抽取方法主要分为三类：基于Seq2Seq的方法、基于填表的方法和基于标记的方法。

(1) 基于Seq2Seq的方法

基于Seq2Seq的方法指：模型通常将三元组视为token序列，将三元组抽取任务转换为按一定顺序生成三元组的生成任务。针对复杂语境下的重叠三元组，Zeng^[25]等人首次提出基于Seq2Seq架构的CopyRE模型，采用拷贝机制，首先生成关系，再生成文本中全部的三元组以解决关系重叠问题。在此基础上，他们还对提取顺序进行研究，提出基于多任务学习的CopyMTL模型^[26]，通过结合强化学习来提取多个词组成的实体。与此同时，Fu^[27]等人和Nayak^[28]等人也对此问题进行研究，Fu等人提出一种基于GCN的方法来提取关系三元组，Nayak等人提出一种基于seq2seq模型，在每个时间步抽取一个词的表示方案。2021年，Ye^[29]等人提出一种基于Transformer的生成式对比学习的关系抽取框架，旨在解决递归神经网络无法捕获长期依赖关系导致生成不可靠的三元组问题。Tian^[30]等

人于 2022 年提出一种让模型去学习句法依赖关系的策略，该策略首先提取出头实体和尾实体，再计算它们的关系，以便提高关系抽取的精度。

(2) 基于填表的方法

基于填表的方法指：模型通常维护与每个关系对应的表，其每一项均指示一个 token 对是否具有对应的关系，即表格中有标记的项通常表示有关系的两个实体的开始和结束位置。2020 年，Wang^[31]等人采用序列编码器和关系表编码器相互交互的方法，结合注意力机制以捕获实体识别和关系抽取任务的交互信息，提高模型性能。同年，为解决暴露偏差问题，Wang^[32]等人提出 TPLinker 模型，将标注框架统一建模为字词对的链接问题，该框架能够解决实体重叠关系和实体的嵌套问题。2021 年，Ren^[33]等人首次在基于填表的方法中考虑字词对和各类关系的全局关联，弥补了仅依赖单一字词对局部特征的局限性。对于目前实体识别和关系识别任务间的特征交互不平衡问题，Yan^[34]等人设计了一个分区过滤网络，用于融合实体识别和关系识别的任务表示，通过共享分区，保证子任务能够进行正确的双向交互，不足的是，该模型采用独立的模块来模拟实体间的交互和实体与关系间的交互。针对 TPLinker 模型中实体与关系间的约束不足问题，Shang^[35]等人于 2022 年将联合抽取转换为一个细粒度的三元组分类问题，从而能一步提取三元组并直接识别三元组的正确性，解决了联合方法中存在的误差传播和信息冗余问题。

(3) 基于标记的方法

基于标记的方法通常指：模型通过标记来预测关系。在这些模型中，二进制标记序列常被用来确定实体的开始和结束位置，有时也被用来确定两个实体之间的关系。Zheng^[36]等人首次提出统一标注的策略，结合偏置损失来增强实体标签的相关性，但未能考虑三元组重叠问题。针对以上问题，Yu^[37]等人提出基于跨度的标注策略，先标记出头实体的位置和类型，再标记尾实体的位置和预测头尾实体间的关系。不足的是，由于采用两个解码器，忽视了头尾实体间的潜在关系。Li^[38]等人将联合关系抽取任务形式化为多轮问答式任务，根据回答模板化的问题来提取三元组，但由于采用问答机制需要对句子进行多次编码，模型的计算复杂度高。Dai^[39]等人提出一种基于词位置的标注策略，通过结合基于位置的注意力机制生成基于源词位置的内容向量。2020 年，Wei^[40]提出一种层

叠指针网络，将关系建模为头实体到尾实体的映射，先提取句中的所有的头实体，再提取每个头实体在所有可能的关系下对应的尾实体，但模型两个子任务的交互不足。Li^[41]等人则参考Wei的思路，改变三元组抽取顺序，先抽取出实体和关系，再抽取出对应的尾实体。为充分挖掘实体识别和关系抽取间复杂的交互作用，Wu^[42]等人采用两个同构的双向类型注意 LSTM 来进行实体类型增强表示和关系类型增强表示并通过交叉类型注意机制来加强实体类型和关系类型之间的相互依赖关系。2021年，Zheng^[43]等人年提出 PRGC 模型，先预测潜在关系，再通过序列标注预测潜在关系下可能的头尾实体，提升模型的计算效率。为充分利用实体与关系的潜在关系，Ren^[44]等人在2022年提出了一个双向的提取框架，该框架先从头实体到尾实体的方向预测出所有可能的实体，再从尾实体到头实体的方向预测出所有可能的实体，以减少实体抽取的遗漏。

1.3 本文研究内容

通过分析总结课题的研究现状，本文发现在面对复杂语境中的重叠关系问题时，联合关系抽取方法存在曝光误差问题和头实体抽取任务与尾实体抽取任务交互不足问题。基于以上观察，本文主要工作如下：

针对联合关系抽取中曝光误差问题，本文提出一种基于全局特征的指针标注联合关系抽取模型。该模型采用指针标注框架，能有效地解决复杂语境中的关系重叠问题。同时，在头实体识别阶段前，根据 BERT 对句子编码的特征向量，采用多头注意力机制和多核卷积神经网络提取文本多粒度的全局特征，将该特征与每个词嵌入进行融合，用于指导头实体预测。这有助于提高头实体预测的精度，拉近头实体预测值与真实值的概率分布，缓解曝光误差问题。

针对头实体抽取任务与尾实体抽取任务交互不足的问题，本文设计一种自适应关系-实体信息融合模块，并提出一种融合关系与实体信息的联合关系抽取模型。该模块主要由一个关系注意力模块和两个门控融合模块组成。其中，关系注意力模块主要对每个词与关系间的交互进行建模，以得到不同关系对每个词的重要程度，即关系嵌入信息。后两个门控模块主要用于将关系嵌入信息和头实体信息与每个词嵌入进行融合，以指导关系和尾实体的预测。该模块充分挖掘了头实体与关系的潜在语义信息，加强了头尾实体抽取任务间的交互。

本文在数据集 NYT 和 WebNLG 上进行了基线模型对比实验、消融实验以及重叠问题的评估实验，以验证本文模型在重叠关系下的性能。实验结果显示，本文提出的最终模型在 NYT 数据集上的 F1 值达到 91.6%，在 WebNLG 数据集上达到 92.6%。实验结果表明，本文提出的模型在解决关系重叠问题上具有明显优势。

1.4 本文组织结构

本文一共分为 6 章，具体结构如下：

第 1 章：绪论。本章首先介绍实体关系的研究背景及意义。然后，对国内外现有的研究展开论述。最后，概述本文的主要研究内容和各章节内容。

第 2 章：相关技术介绍。本章详细介绍本文模型涉及到的关键技术。主要包括全连接神经网络、卷积神经网络、自注意力机制和预训练语言模型 BERT。

第 3 章：基于全局特征的指针标注联合关系抽取模型。主要对基于全局特征的指针标注联合关系抽取模型做详细介绍。其中包括模型的问题定义、模型的总体架构及各组成模块，例如：句子编码层、全局特征处理层、头实体识别层和关系-尾实体识别层。

第 4 章：基于关系增强的指针标注联合关系抽取模型。主要对基于关系增强的指针标注联合关系抽取模型做详细介绍。重点介绍模型的自适应关系-实体融合模块和关系-尾实体标注器，最后，介绍模型的训练流程、伪代码实现和模型的优化算法。

第 5 章：实验设计与分析。主要介绍模型的实验环境以及模型的对比实验，并对实验结果进行分析。实验环境其中包括：数据集的详细介绍、实验评价方法、模型的实验环境和超参数设置。模型的对比实验包括：模型与基线模型的对比、模型的消融实验以及模型在重叠问题上的实验评估。

第 6 章：总结与展望。主要概括本文工作并对未来研究方向进行展望。

第2章 相关技术介绍

本章对基于指针标注的联合关系抽取任务中涉及到的相关技术进行介绍，主要包括全连接神经网络、卷积神经网络、注意力机制和预训练语言模型BERT。

2.1 全连接神经网络

全连接神经网络（Fully connected neural network, FCNN）是一种基础的深度学习神经网络。FCNN 的目标是拟合一个目标函数，这函数被称为特征到标签的映射，通过多次计算简易的非线性函数，可以将输入空间的特征向量映射到输出空间。

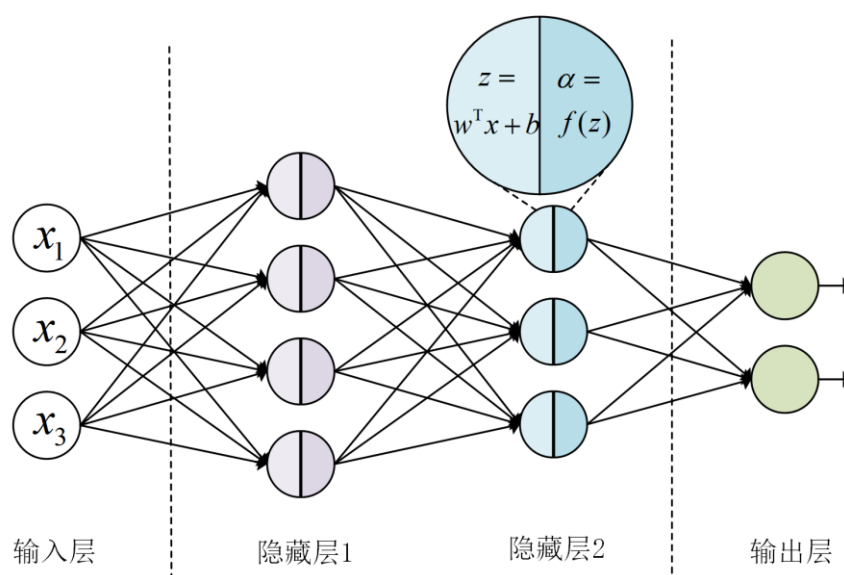


图 2.1 全连接神经网络

一个四层的 FCNN 如图 2.1 所示。FCNN 主要由一层输入层、若干个层隐藏层和一层输出层构成。其工作原理是将输入数据传递到隐藏层并逐层进行计算，最终由输出层输出结果，此过程被称为正向传播过程。其中，每个隐藏层由一组带权重信息和偏置信息的神经元构成，每个神经元均与前一层的所有神经元相连，并对前一层的输出数据进行加权求和，再由激活函数得到非线性输出。每个隐藏层中的单个神经元计算公式如下所示：

$$h_i^l = f\left(\sum_j^n w_{ij} h_j^{l-1} + b_i\right) \dots\dots\dots (2.1)$$

其中， h_j^{l-1} 表示上一层隐藏层的输出， b_i 表示偏置信息， w_{ij} 表示当前层神经元与上一层神经元间的权重， $f(\cdot)$ 表示激活函数。FCNN 中的激活函数均是非线性函数，能够增强 FCNN 模型的表达能力和非线性能力，解决线性不可分问题。在 FCNN 中，常用的激活函数有 sigmoid(\cdot)、tanh(\cdot) 和 ReLU(\cdot) 等，计算公式如下所示：

$$\text{sigmoid}(x) = \frac{1}{1 + e^{-x}} \dots\dots\dots (2.2)$$

$$\text{tanh}(x) = \frac{e^{2x} - 1}{e^{2x} + 1} \dots\dots\dots (2.3)$$

$$\text{ReLU}(x) = \begin{cases} x, & \text{if } x \geq 0 \\ 0, & \text{if } x < 0 \end{cases} \dots\dots\dots (2.4)$$

FCNN 通过误差的反向传播对网络权重进行调整、优化，从而降低损失函数值，使预测结果值趋近于实际值。虽然 FCNN 的参数量大，容易出现过拟合的问题，且对空间结构的平铺降维信息表达性差，但鉴于结构简单，不要求输入尺寸为固定尺寸，能聚合数据信息，目前依然是大多数自然语言处理任务中常用的神经网络之一。

2.2 卷积神经网络

卷积神经网络(Convolutional Neural Network, CNN)在关系抽取中通常采用 LeCun^[45] 等人提出的卷积层-池化层这一经典结构。与传统的 FCNN 不同，单个 CNN 的神经元可以响应其覆盖范围内的神经单元的一部分，这种网络结构由反向传播训练后能更准确地提取特征。CNN 由输入层、卷积层、池化层和全连接层组成。因为文本数据的序列特性不同于图像数据的序列特性，因此相比于计算机视觉领域最常用的二维 CNN，一维 CNN 在 NLP 领域能发挥更大的作用，本文只介绍与 NLP 相关的一维 CNN。具有不同卷积核的 CNN 对文本向量的计算流程，如图 2.2 所示。

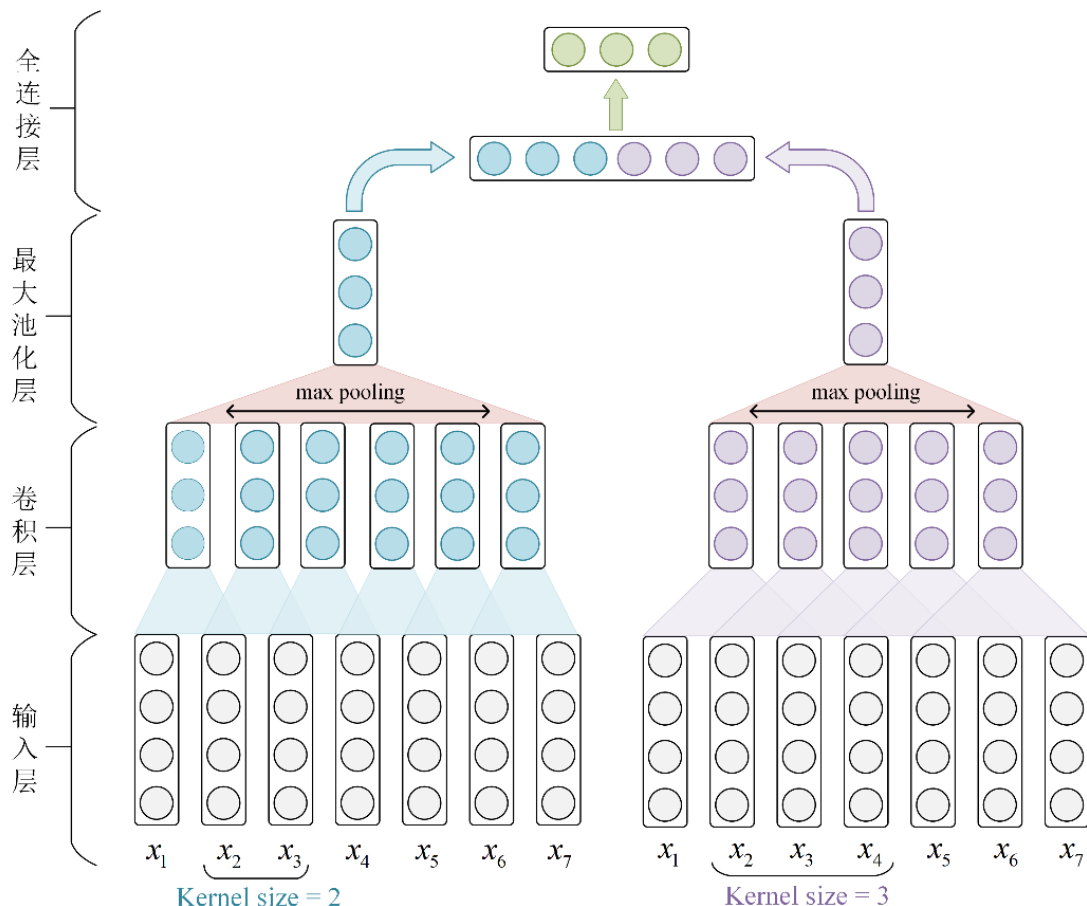


图 2.2 不同卷积核的 CNN 对文本向量的计算流程

如图 2.2 所示，CNN 输入层的主要作用是接收预处理后的文本序列 $X = \{x_1, x_2, x_3, \dots, x_n\}$ ， $X \in \mathbb{R}^{n \times d}$ 。其中， n 代表输入序列长度， d 代表词嵌入维度， $n = 7$ ， $d = 4$ ，句子的维度为 $n \times d$ 。

卷积层的作用是对输入序列 X 进行卷积运算，采用卷积核提取序列 X 的局部特征。值得注意的是，为保证每个词向量的语义完整性，一般将卷积核的列数设置为词向量的维度数，以确保卷积窗口仅在序列的长度方向上滑动。如图 2.2 所示，CNN 采用大小为 2 和 3 的卷积核对序列 $X = \{x_1, x_2, x_3, \dots, x_n\}$ 进行卷积运算，计算公式为：

$$m_i = g(W * X_{ii+k-1}) + b \dots \dots \dots (2.5)$$

其中， $X_{ii+k-1} \in \mathbb{R}^{k \times d}$ 表示序列 X 的第 i 个位置到第 $i+k-1$ 的向量， $g(\cdot)$ 表示激活函数， $W \in \mathbb{R}^{k \times d}$ 表示卷积核的矩阵数值。卷积核在输入序列 X 上计算，得到向量 $m = [m_1, m_2, \dots, m_{n-k+1}]$ 。

池化层的作用是在池化窗口中筛选出具有代表性的特征，过滤掉无用或者作用不明显的特征，减少特征的维度。常用的池化操作有：最大池化，取池化窗口中的最大值作为采样值；平均池化，取池化窗口中的加权平均值作为采样值；分段池化，通常适用于关系抽取任务，采用两个命名实体将句子分为三个部分，分别对头实体前面部分，实体对中间部分，尾实体后面部分进行池化。

全连接层的作用是：首先，将经过多次卷积计算和池化计算后的序列进行拼接，然后输入到分类器中，并输出分类概率。

2.3 注意力机制

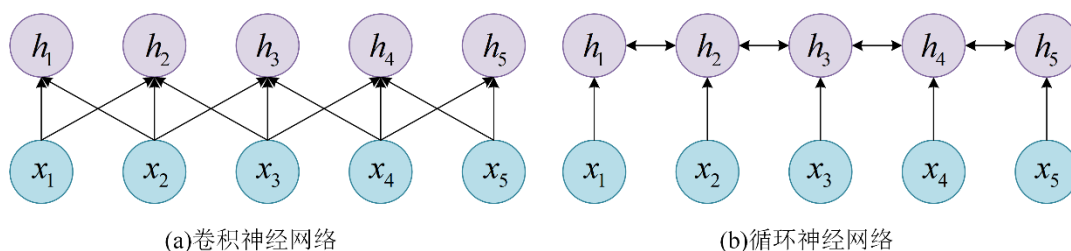


图 2.3 卷积神经网络和双向循环网络处理序列

在自然语言处理中，通常使用卷积模型或序列模型来获取序列的全局信息。如图 2.3 所示，无论卷积模型还是序列模型均是对序列的一种局部编码。其中，卷积模型是基于 N-gram 的局部编码，通过正方形或者矩形的卷积核获取局部感受野，会造成关联信息的缺失；序列模型，由于处理长序列数据会造成梯度消失等问题，只能建立短距离依赖。

为解决以上问题，在 2017 年，Google 团队提出注意力机制^[46]，通过注意力机制来处理序列，不需要通过多个卷积核来获取全局信息，也不需要逐步递归获取全局信息，直接一步到位获取全局信息。

如图 2.4 所示，注意力机制的核心思想可描述为：一个查询到一组键值对的映射，输入查询 *Query*，根据 *Key* 和 *Value* 得到 *Attention Value*，其计算公式为：

$$\text{Attention}(\text{Query}, \text{Source}) = \sum_{i=1}^{L_x} \text{Similarity}(\text{Query}, \text{Key}_i) * \text{Value}_i \dots (2.6)$$

其中，*Source* 表示一组 $\langle \text{Key}, \text{Value} \rangle$ 元素的键值对，*Value* 表示 *Key* 对应的权重，*Query* 表示输入的词向量， L_x 表示键值对 $\langle \text{Key}, \text{Value} \rangle$ 的数量。

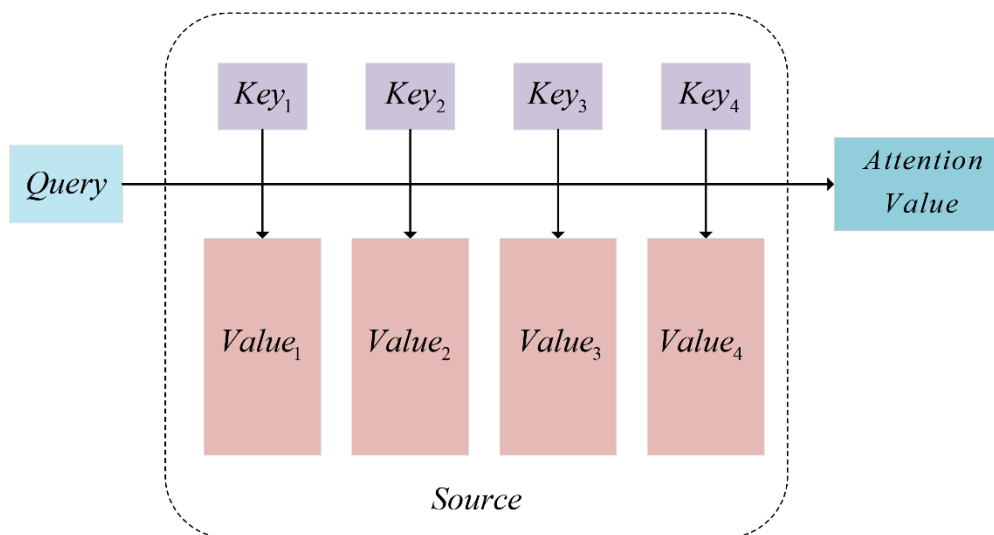


图 2.4 注意力机制的思想

注意力机制计算可拆解为三个阶段：第一阶段，查询值与一组键值的相似性计算；第二阶段，对相似度进行归一化处理；第三阶段，加权求和。如图 2.5 所示。

第一阶段，查询值与一组键值的相似性计算。常见的计算注意力相关性的 Similarity 函数计算公式如下：

$$\text{Similarity}(\text{Query}, \text{Key}_i) = \text{Query} \cdot \text{Key}_i \dots\dots\dots (2.7)$$

$$\text{Similarity}(\text{Query}, \text{Key}_i) = \frac{\text{Query} \cdot \text{Key}_i}{\|\text{Query}\| \cdot \|\text{Key}_i\|} \dots\dots\dots (2.8)$$

$$\text{Similarity}(\text{Query}, \text{Key}_i) = \text{MLP}(\text{Query}, \text{Key}_i) \dots\dots\dots (2.9)$$

其中，Query 表示输入的词向量， Key_i 表示键值对的一个键，公式(2.7)使用向量点积的方式求 Query 和 Key_i 的相似度，公式(2.8)采用计算 Query 和 Key_i 余弦值的方式求相似度，公式(2.9)采用神经网络的方式求 Query 和 Key_i 的相似度。

第二阶段，对相似度进行归一化处理。计算公式如下：

$$a_i = \text{softmax}(\text{Similarity}_i) = \frac{e^{\text{Similarity}_i}}{\sum_{j=1}^{L_x} e^{\text{Similarity}_j}} \dots\dots\dots (2.10)$$

其中，Softmax 为将结果归一化到[0,1]区间的激活函数， L_x 为键值对的数量。

第三阶段，加权求和。计算公式如下：

$$\text{Attention}(\text{Query}, \text{Source}) = \sum_{i=1}^{L_s} a_i \cdot \text{Value}_i \dots\dots\dots (2.11)$$

注意力机制的 Query 和 Source 分别来自不同的句子，例如在机器翻译模型中，Query 一般来自目标语言，Source 一般来自源语言。自注意力机制是注意力机制的特殊情况，其 Query 和 Source 都来自同一句子，计算公式如下：

$$\text{SelfAttention}(Q, K, V) = \text{softmax}\left(\frac{QK^T}{\sqrt{d_k}}\right)V \dots\dots\dots (2.12)$$

其中，Q 为 Query，K 和 V 是 Source 的键值对， $\sqrt{d_k}$ 为缩放因子。

多头自注意力机制是利用不同的 W_Q 、 W_K 、 W_V 得到不同空间下的注意力值 head，最后将这些 head 进行拼接，计算公式如下：

$$\text{head}_i = \text{SelfAttention}(Q_i, K_i, V_i) = \text{SelfAttention}(SW_Q^i, SW_K^i, SW_V^i) \dots\dots (2.13)$$

$$\text{MultiHead} = \text{Concat}(\text{head}_1, \text{head}_2, \dots, \text{head}_h)W \dots\dots\dots (2.14)$$

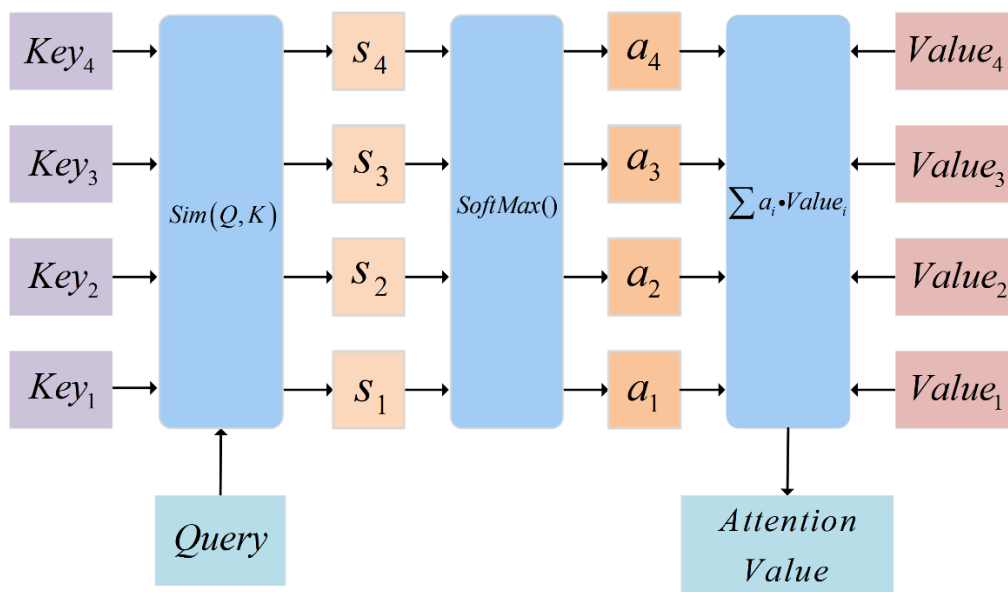


图 2.5 注意力机制的计算流程

2.4 预训练语言模型 BERT

2018 年，Google 团队提出 BERT^[47] (Bidirectional Encoder Representations from Transformers) 模型。BERT 的结构如图 2.6 所示，该模型不同于由 Transformer 的解码器构成的 GPT 模型，BERT 模型由 Transformer 的编码器构成，

采用基于掩码的语言模型进行训练，从而得到蕴含上下文深层语义信息的词向量表示。

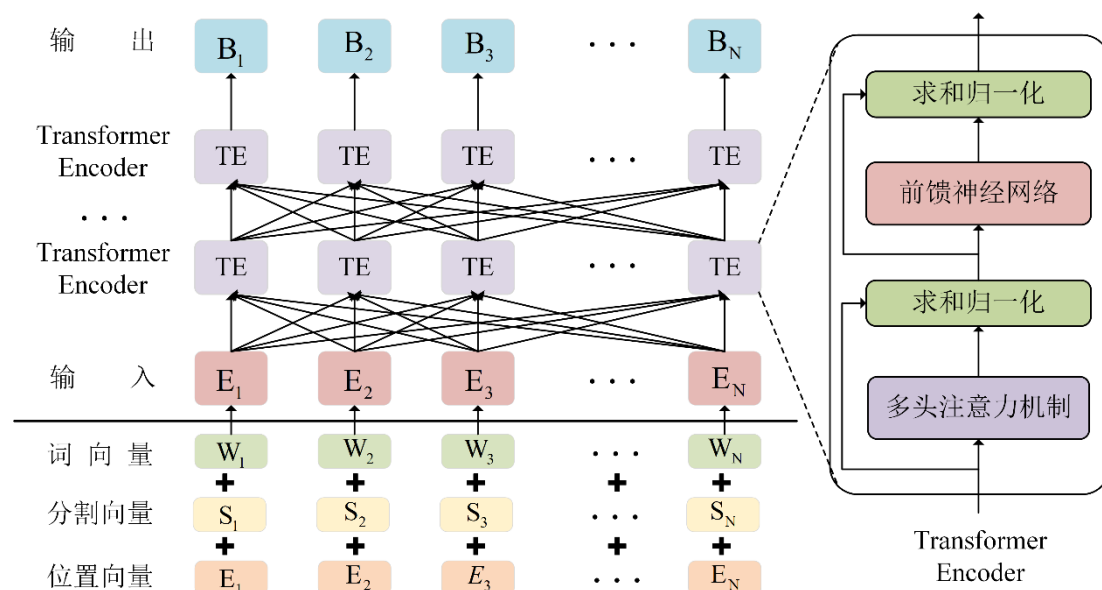


图 2.6 BERT 的结构

BERT 的输入包含三部分：单词的词向量、位置向量以及分割向量。与此同时，BERT 还引入了[CLS]、[SEP]两个特殊字符。[CLS]通常加在输入文本的开头，表示整段文本的概要信息；[SEP]用来当作不同句子的分隔符或者代表句子的结束。

根据 BERT 的不同的规模，Devlin^[47]等人推出了 BERT 的两个版本：BERT base 和 BERT large。其中，BERT base 版本主要由 12 层的 Transformer 编码器堆叠，隐藏向量维度为 768，每个 Transformer 编码器内部的多头注意力共有 12 个头；BERT large 版本主要由 24 层的 Transformer 编码器堆叠，隐藏向量维度为 1024，每个 Transformer 编码器内部的多头注意力共有 16 个头。本文采用的 BERT 版本为最小的 BERT base 模型。

2.5 本章小结

本章主要介绍了本文课题研究涉及的相关技术。其中包括：全连接网络的内部结构和作用，卷积神经网络对文本向量的计算流程，注意力机制和多头注意力机制的结构特点及计算流程，预训练语言模型 BERT 的架构及作用。为本文的模型和方法奠定了基础。

第3章 基于全局特征的指针标注联合关系抽取

在当前解决实体关系重叠的联合关系抽取任务中，存在训练和推理的搜索空间不一致的问题，即曝光误差问题。针对这一问题，本章提出一种基于全局特征的指针标注联合关系抽取模型（Joint Relation Extraction Model based on Pointer Annotation with Global Features, JREPA_GF）。首先，本章详细介绍抽取任务及重叠关系三元组的问题定义；其次，介绍 JREPA_GF 模型的提出动机和整体框架；最后，对 JREPA_GF 模型的各个部分进行详细阐述。

3.1 问题定义

在本文中，实体关系抽取的目标为自动且高效地抽取所有可能的（“头实体”，关系，“尾实体”）关系三元组。

本文将关系抽取任务形式化为：给定一条包含 L 个单词的非结构化句子 $X = \{x_1, x_2, \dots, x_L\}$ ，其中， x_i 代表句子中第 i 个位置的单词，建模的目标是识别出句子 X 中所有可能存在的关系三元组集合 $Y = \{Q_1, Q_2, \dots, Q_N\}$ 。其中， $Q_i = (s_i, r_i, o_i)$ ， $1 \leq i \leq N$ ， N 为三元组的数量， Q_i 代表句子的第 i 个三元组。具体地： s_i 、 o_i 和 r_i 分别代表第 i 个关系三元组中的头实体、尾实体及关系。头实体和尾实体源于句子 X 的实体集合 $E = \{e_1, e_2, \dots, e_k\}$ ， k 表示实体的数量，关系源于预先定义的关系类别集合 $R = \{R_1, R_2, \dots, R_m\}$ ， m 表示集合中的关系种类数目。需要注意的是，在三元组之间可能存在共享的实体和关系。

本文将关系抽取任务建模为：头体实体到尾实体的映射函数。首先采用指针标注策略来标注句子中的头实体。然后，根据标注出的头实体信息在特定的关系下，抽取出尾实体和关系。本文将关系抽取任务定义为以下形式：

$$P(Y|X) = \prod_{i=1}^N p(Q_i|X) \dots\dots\dots (3.1)$$

$$= \prod_{i=1}^N p((s_i, r_i, o_i)|X) \dots\dots\dots (3.2)$$

$$= \prod_{i=1}^k p(s_i|X) \prod_{i=1}^m p(r_i|s_i, X) \prod_{i=1}^{k \times m} p(o_i|s_i, r_i, X) \dots\dots\dots (3.3)$$

本文的目标为使最大化生成的概率 P 的似然，其中， $p(s_i|X)$ 表示抽取的第 i 个三元组中头实体概率， $p(r_i|s_i, X)$ 表示实体 s_i 存在关系 r_i 的概率， $p(o_i|s_i, r_i, X)$ 表示实体 s_i 在关系 r_i 存在尾实体 o_i 的概率。

重叠关系问题通常指：在复杂的语义场景中，句子中三元组的头实体或尾实体可能会产生重叠。这意味着，三元组中的实体对可能存在多种关系，或者三元组中的一个实体与另一三元组中的实体间存在多种关系，从而影响关系抽取性能。

类别	句子	三元组关系对
常规实体 (Normal)	Suzhou is located in China. country	(“Suzhou”, “country”, “China”)
单实体重叠 (SEO)	Xiao Ming was born in Beijing, the capital of China. birthplace, capital	(“Xiao Ming”, “birthplace”, “Beijing”) (“China”, “capital”, “Beijing”)
实体对重叠 (EPO)	Miss Wang is Xiao Ming's mother and teacher. teacher, mother	(“Miss Wang”, “mother”, “Xiao Ming”) (“Miss Wang”, “teacher”, “Xiao Ming”)

图 3.1 实体重叠类型示例

如图 3.1 所示，本文针对句子实例中的不同重叠关系情况，将句子实例划分为三种类别：（1）正常类（Normal）；（2）单实体重叠类（Single Entity Overlap, SEO）；（3）实体对重叠类（Entity pair Overlap, EPO）。具体而言，Normal 类指句子实例中未出现三元组实体重叠的情况；SEO 类指句子实例中至少出现一个实体属于多个不同的三元组的情况。例如，在 SEO 类的句子中，实例的实体“Beijing”属于两个不同关系的三元组：（“Xiao Ming”，birthplace，“Beijing”）和（“China”，capital，“Beijing”）；EPO 类指句子实例中的实体对至少出现两种不同关系的情况，例如，在 EPO 类的句子中，存在实体对“Miss Wang”和“Xiao Ming”属于两种不同的关系，分别属于（“Miss Wang”，mother，“Xiao Ming”）和（“Miss Wang”，teacher，“Xiao Ming”）两个不同的三元组。

3.2 JREPA_GF 模型

本节将对基于全局特征的指针标注联合关系抽取模型 (JREPA_GF 模型), 进行详细介绍。提出 JREPA_GF 模型的动机是: 当前联合关系抽取模型在处理重叠关系三元组时, 普遍存在严重的曝光误差现象。即在训练和推理时使用的输入不同, 训练时的输入来源于真实样本, 而推理时的输入来源于模型的上一步输出, 影响模型的泛化性能。

如图 3.2 所示, JREPA_GF 模型的架构图描述了句子关系抽取的过程。该模型主要分为四个部分: 句子编码层、全局特征处理层、头实体识别层和关系-尾实体识别层。JREPA_GF 模型的创新之处在于: 在句子编码层后, 融合了本文提出的基于 BERT 的多粒度的全局特征, 丰富了先验特征的语义表达, 提高头实体抽取的准确率, 从而缩小训练空间与推理搜索空间的差距, 缓解曝光误差。同时, 本文采用指针标注框架, 能一次性抽取出多个三元组并解决重叠关系问题, 增强实体关系抽取模型的性能表现。

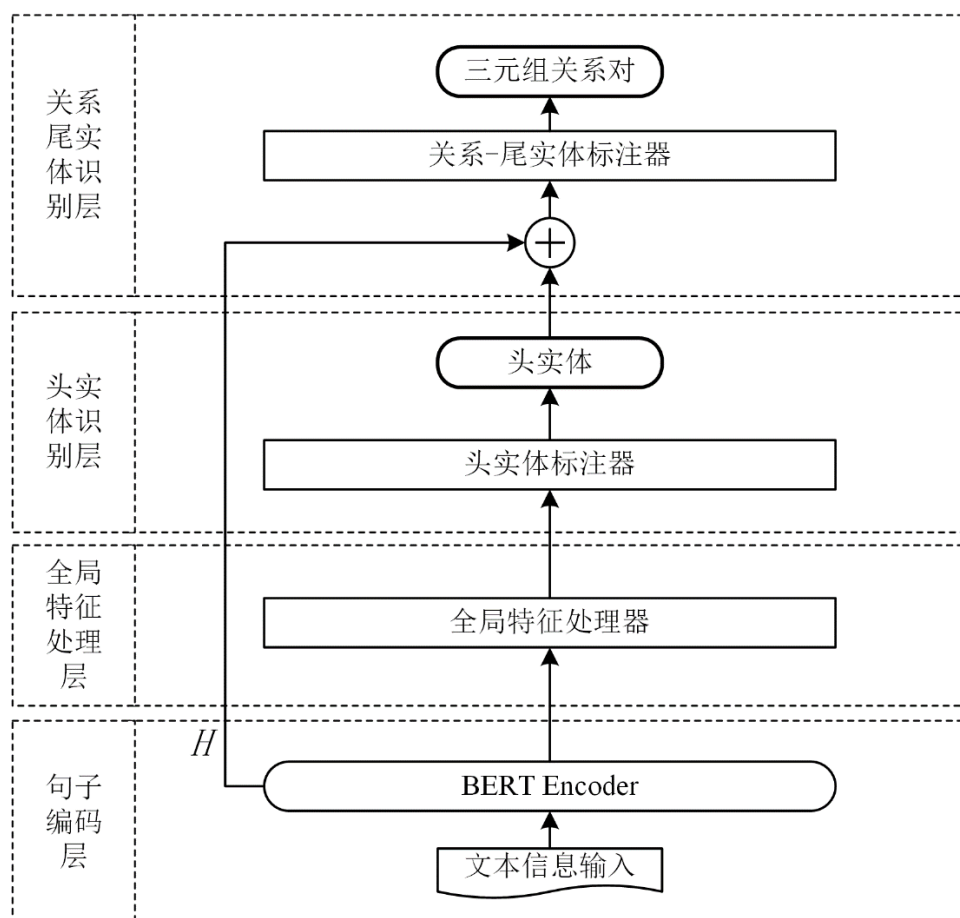


图 3.2 JREPA_GF 模型的结构

如图 3.2 所示, JREPA_GF 模型的句子编码层采用 BERT 来获取文本信息中每个单词的词嵌入表示, 使其融合上下文的深层语义信息。全局特征处理层对 BERT 得到的句子语义信息进行处理, 并得到多粒度、多层次的全局语义特征信息的词嵌入向量, 丰富先验特征的语义表达。实体识别层采用头实体标注器, 将全局特征处理层得到的具有丰富先验知识的全局特征, 用于识别句子中所有可能存在的头实体。关系-尾实体识别层采用关系-尾实体标注器, 针对标记的头实体, 识别出所有可能存在与其相关的尾实体及头尾实体的关系。其中, 关系-尾实体标注器由一组特定关系下的尾实体标注器组成, 与头实体标注器具有相同的结构。

3.2.1 句子编码层

句子编码层的目标是采用 BERT 对目标文本进行编码。BERT 作为预训练语言模型, 由多层相同结构的 Transformer 的编码器组成, 通过采用基于掩码的训练方式学习词的上下文语义信息, 从而获取蕴含上下文深层语义信息的词向量表示。在其输出结果上添加简单的网络架构, 便能在自然语言处理的下游任务(如序列标注任务、文本分类任务以及句对关系判断任务等)中取得优异的表现。因此, 本文将其应用于实体关系抽取任务。在本文中采用的 BERT 模型由 12 层的 Transformer 的编码器组成。

设 BERT 的输入为序列 $X = \{x_1, x_2, x_3, \dots, x_N\}$, $X \in \mathbb{R}^{N \times d}$, N 表示序列长度, x_i 表示每个词的独热向量表示与相应位置信息嵌入表示的总和, 编码层的计算公式如下所示:

$$H = \text{BERT}(X) \dots\dots\dots (3.4)$$

$$\{h_1, h_2, h_3, \dots, h_N\} = \text{BERT}(\{x_1, x_2, x_3, \dots, x_N\}) \dots\dots\dots (3.5)$$

其中, $H = \{h_1, h_2, h_3, \dots, h_N\}$ 表示 BERT 的输出, $H \in \mathbb{R}^{N \times d_h}$, N 表示 BERT 输出句子序列的长度, d_h 表示每个 x_i 对应输出融合上下文语义信息的词嵌入表示 h_i 的维度。

3.2.2 基于 BERT 的全局特征处理层

传统的联合关系抽取模型，通常采用实体的类型特征、命名实体特征、词性标注特征和位置特征等局部特征进行特征补充，但往往忽略句子全局特征的重要性。此外，传统的句子的语义特征信息提取方式，通常采用单一的 CNN 或 RNN 等神经网络，但这些方法往往存在一定的局限性。CNN 通过一个卷积核只能获取到序列信息的局部感受野，尽管通过堆叠多个卷积核能扩展局部感受野，捕获全局特征信息，然而，在池化操作过程中，实际上会产生信息损失；RNN 容易产生梯度消失问题，只能提取较短距离的句子特征，因此也会信息损失。为解决以上问题，本文提出一种基于 BERT 的全局特征。其处理流程如图 3.3 所示。

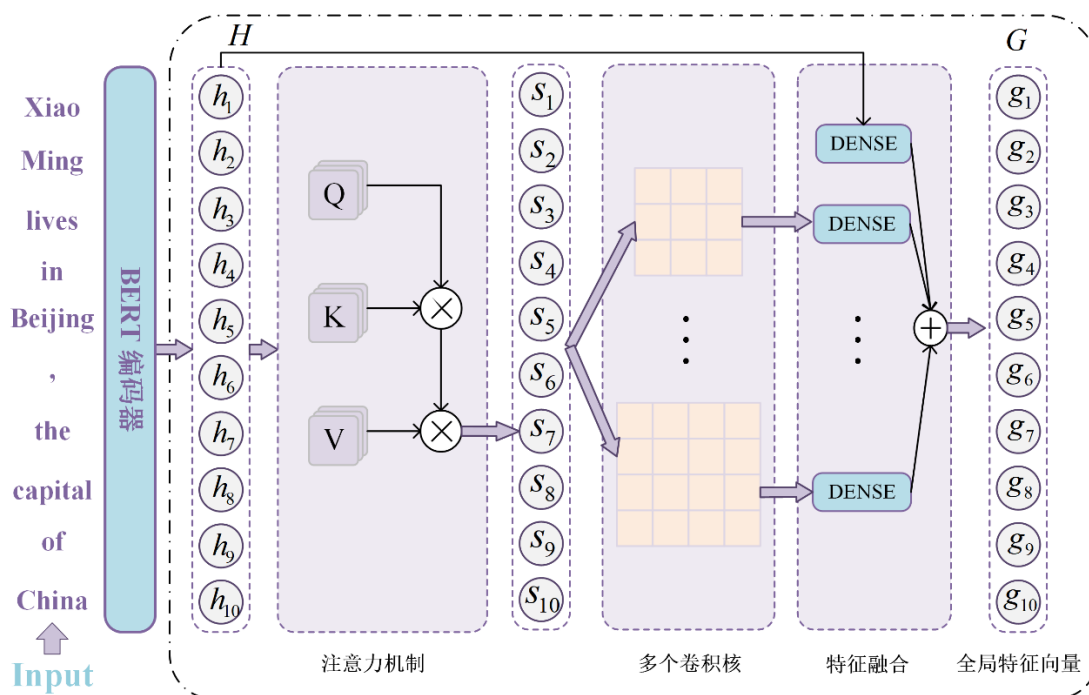


图 3.3 全局特征处理流程

基于 BERT 的全局特征处理流程，如图 3.3 所示。首先，获取句子编码层输出的、具有深层上下文语义信息的句子嵌入表示 H 。其次，采用多头自注意力机制来计算句子在多个向量空间内每一个词与其余词的相关性，加强相关性高的词之间的影响，弱化无关噪声词对句子表示的影响。然后，采用卷积神经网络对上述输出向量进行向量的特征提取与过滤。值得注意的是，为获取多粒度、不同层次的特征信息表示，本文采用多个不同大小的卷积核进行卷积运算。

最后将提取出的特征向量表示与 BERT 得到的句子嵌入表示 H 分别进行线性变换，再进行拼接操作，最终得到本文的基于 BERT 的全局特征向量。通过注意力机制处理在前，卷积处理在后，能增强卷积获取全局相关性的能力。

下面介绍全局特征的计算过程，一共分为三个步骤：首先是多头注意力机制的计算，其次是多核卷积神经网络的计算，最后是特征向量的拼接计算。

(1) 多头自注意机制计算

为提取文本信息中多粒度、多层次的特征信息，多头注意力机制的思想与 CNN 中采用多个卷积核获取文本信息的多元特征类似。自注意力机制能计算出句子内每个词和其他所有词之间的语义相关性，强化相关性强的词之间的影响，同时过滤掉无关噪声词对句子的影响。然而，自注意力机制仅在单一子空间提取关键信息。相比之下，多头注意力机制由多个结构相同的自注意力模块组成，能够在多个向量子空间中提取输入序列中不同层次、不同粒度的特征。

多头注意力机制使用 h 组随机初始化生成的查询矩阵 W_Q 、值矩阵 W_V 以及键矩阵 W_K 集合，与输入序列 $H = \{h_1, h_2, h_3, \dots, h_N\}$ ， $H \in \mathbb{R}^{N \times d_h}$ 相乘，得到 h 组向量集合 $T = \{T_1, T_2, \dots, T_h\}$ ， $T_i = (Q^i, K^i, V^i)$ ，以达到将输入词向量投影到不同的表示子空间的效果。 T_i 表示第 i 个经过初始化生成的查询向量 Q^i 、键向量 K^i 和值向量 V^i 。每个不同的表示子空间称为一个 head，在第 i 个 head 中进行注意力计算的公式如下：

$$Q^i = HW_Q^i \dots\dots\dots (3.6)$$

$$K^i = HW_K^i \dots\dots\dots (3.7)$$

$$V^i = HW_V^i \dots\dots\dots (3.8)$$

$$\begin{aligned} head_i &= \text{SelfAttention}(Q^i, K^i, V^i) \\ &= \text{SelfAttention}(HW_Q^i, HW_K^i, HW_V^i) \\ &= \text{Softmax}\left(\frac{Q^i (K^i)^T}{\sqrt{d_k}}\right) V^i \dots\dots\dots (3.9) \\ &= \text{Softmax}\left(\frac{HW_Q^i H^T W_K^{iT}}{\sqrt{d_k}}\right) HW_V^i \end{aligned}$$

其中, $H = \{h_1, h_2, h_3, \dots, h_N\}$, $H \in \mathbb{R}^{N \times d_h}$ 表示多头注意力的输入序列; W_Q^i 、 W_K^i 、 $W_V^i \in \mathbb{R}^{d_h \times d'}$ 分别表示第 i 个 head 子空间中的查询矩阵、键矩阵以及值矩阵, $head_i \in \mathbb{R}^{N \times d'}$; $Q^i(K^i)^T$ 表示第 i 个 head 查询向量 Q^i 与键值向量 K^i 采用点积操作进行相似度计算, 得到输入序列与内容的相似度; Softmax 函数为归一化函数, 将数值映射到 0 和 1 区间, 且数值的和为 1; $\sqrt{d_k}$ 为调节 $Q^i(K^i)^T$ 点积结果的因子, 防止点积结果过大影响 Softmax 函数归一化的作用。

通过在 h 个子空间进行自注意力计算, 得到所有 $head_i$, 将所有的 $head_i$ 进行拼接, 再经过参数矩阵 W_c 进行线性运算, 得到多层次且具有丰富特征信息的句子隐层表示 S , 其计算公式如下所示:

$$S = \text{Concat}(head_1, head_2, \dots, head_h)W_c \dots \dots \dots (3.10)$$

其中, $head_i \in \mathbb{R}^{N \times d'}$, $\text{Concat}(head_1, head_2, \dots, head_h) \in \mathbb{R}^{N \times d^*h}$, $W_c \in \mathbb{R}^{d^*h \times d'}$, $S = \{s_1, s_2, s_3, \dots, s_N\}$, $S \in \mathbb{R}^{N \times d'}$, $s_i \in \mathbb{R}^{d'}$ 。

(2) 多卷积核计算

在经过多头注意力机制得到多层次的句子特征表示 $S = \{s_1, s_2, s_3, \dots, s_N\}$ 后, 使用卷积神经网络对 S 进行卷积操作, 过滤掉 S 中一些无关向量, 使模型聚焦在重要信息上, 得到 S 中不同位置的局部特征。在进行卷积计算时, 采用多个窗口进行独立计算以实现并行化处理。卷积神经网络的计算公式如下:

$$c_i = \text{Relu}(W \cdot S_{ii+w-1} + b), 1 \leq i \leq N - w + 1 \dots \dots \dots (3.11)$$

$$\text{Relu}(x) = \begin{cases} 0, & (x < 0) \\ x, & (x \geq 0) \end{cases} \dots \dots \dots (3.12)$$

其中, $S = \{s_1, s_2, s_3, \dots, s_N\}$, 为卷积神经网络的输入, $S \in \mathbb{R}^{N \times d'}$, $s_i \in \mathbb{R}^{d'}$, $S_{i,j}$ 表示 s_i 到 s_j 的拼接矩阵, $W \in \mathbb{R}^{w \times d'}$ 表示卷积操作的卷积核, w 表示卷积核滑动窗口大小。卷积运算即卷积核 W 与序列 s_{ii+w-1} ($i = 1, 2, \dots, n - w + 1$) 依次做点积运算, 对于超过句子最大长度的部分, 用 0 填充。然后, 通过非线性激活函数 Relu, 将卷积运算后的特征值映射到非线性空间, 以学习到更丰富的特征信息, 获得比单线性模型更强大的表示能力。

此外，为了提取多粒度的句子特征信息，本文采用三个卷积核进行卷积运算。设卷积核集合为： $W = \{w_1, w_2, w_3\}$ ，卷积核的大小分别为 k_1, k_2, k_3 。其中， $k_1 = 3, k_2 = 4, k_3 = 5$ 。具体计算公式如下：

$$a_j = \text{Relu}(w_1 s_{j:j+k_1-1} + b_1) \dots\dots\dots (3.13)$$

$$c_j = \text{Relu}(w_2 s_{j:j+k_2-1} + b_2) \dots\dots\dots (3.14)$$

$$d_j = \text{Relu}(w_3 s_{j:j+k_3-1} + b_3) \dots\dots\dots (3.15)$$

将上述三个卷积核在序列 S 上滑动，分别得到三个不同层次信息的特征向量序列： $A = \{a_1, a_2, a_3, \dots, a_N\}$ 、 $C = \{c_1, c_2, c_3, \dots, c_N\}$ 和 $D = \{d_1, d_2, d_3, \dots, d_N\}$ 。

(3) 特征向量的拼接计算

首先，将蕴含上下文语义信息的词嵌入向量 H 通过全连接层映射到一维子空间中，得到具有实体分类信息的词嵌入信息。其计算公式如下：

$$H' = \text{Relu}(W_1 H + b_1) \dots\dots\dots (3.16)$$

接着，对序列 A 、 C 和 D 分别进行线性变换，通过共享网络的隐藏表示，能综合序列的交互信息，从而得到多粒度的全局信息。计算公式如下：

$$A' = \text{Relu}(W_2 A + b_2) \dots\dots\dots (3.17)$$

$$C' = \text{Relu}(W_3 C + b_3) \dots\dots\dots (3.18)$$

$$D' = \text{Relu}(W_4 D + b_4) \dots\dots\dots (3.19)$$

其中， $H' \in \mathbb{R}^{N \times 1}$ ， $A', C', D' \in \mathbb{R}^{N \times d_c}$ 。

最后，将向量 H' 与向量 A' 、 C' 和 D' 进行拼接，将句子中每个词的词嵌入信息叠加由注意力机制和不同卷积核得到的不同粒度的全局特征，得到具有多粒度、多层次的全局特征向量的句子嵌入信息 G ，以对头实体进行预测。

计算流程如下：

$$G = \text{Concat}(A', B', C', H') \dots\dots\dots (3.20)$$

其中， $G = \{g_1, g_2, g_3, \dots, g_N\}$ ， $G \in \mathbb{R}^{N \times d_g}$ ， N 表示句子的序列长度， d_g 表示每个词的嵌入长度。

3.2.3 头实体识别层

头实体识别层的主要目标是根据全局特征处理层得到的全局向量特征 G ，预测句子中所有潜在的头实体。考虑到句子中可能存在由多个单词组成的长尾实体，例如，由两个单词组成的长尾实体：“Xiao Ming”，本文采用基于指针标注的方式对头实体进行标注。具体而言，将头实体的起始位置的标注问题建模成二分类问题，采用基于二进制分类器的头实体标注器分别标注出头实体的开始和结束位置。头实体识别层的整体架构，如图 3.4 所示。

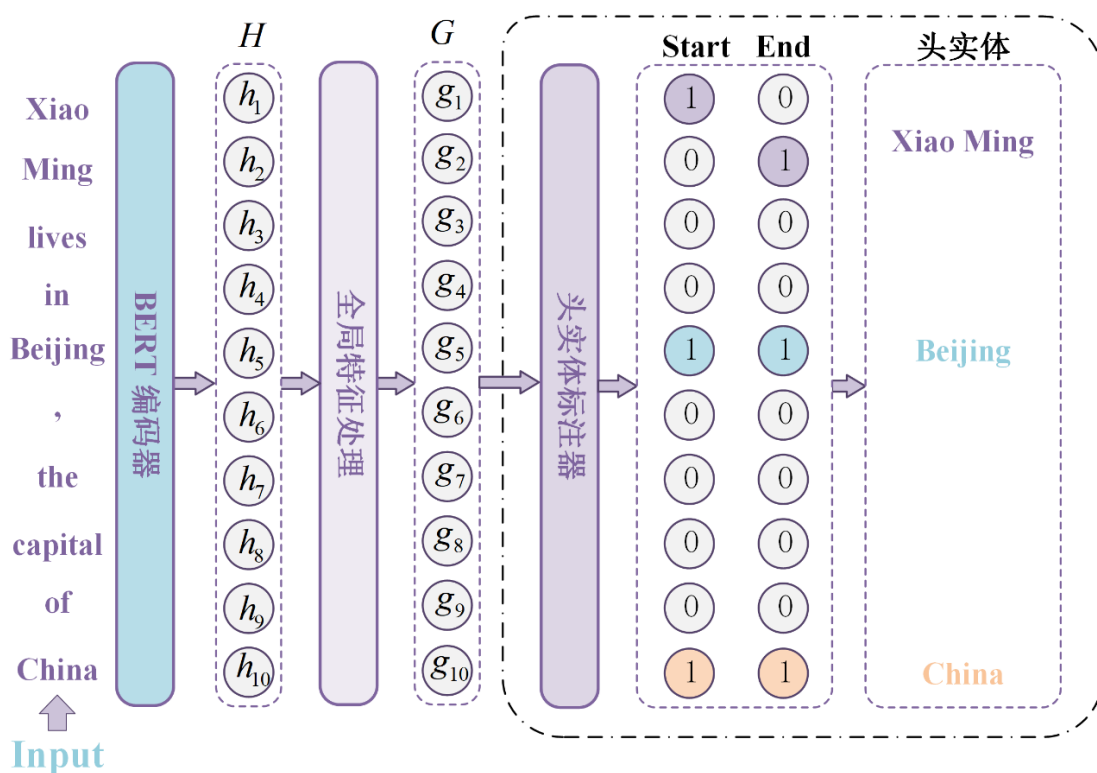


图 3.4 头实体识别层结构

如图 3.4 所示，将目标句子：“Xiao Ming lives in Beijing, the capital of China.”通过句子编码层和全局特征处理层得到全局向量 G 。接着，头实体标注器根据全局向量 G ，标注出头实体的开始标志位与结束标志位。例如，头实体标注器将“Xiao”，“Beijing”，“China”标注为开始位置，将“Ming”，“Beijing”，“China”标注为结束位置。本文按照近邻匹配原则进行头实体检测，即头实体标注器标注头实体的第一个开始位置 Start 与最近第一个结束位置 End 间所有词构成第一

以上内容仅为本文档的试下载部分，为可阅读页数的一半内容。如要下载或阅读全文，请访问：<https://d.book118.com/235104311041011114>