中图分类号：　　单位代号：10280
密　级：　　学　号：XXX

上海大学

硕士学位论文

SHANGHAI UNIVERSITY
MASTER'S DISSERTATION

题目	基于深度学习的认知语言学转喻识别方法研究

作　者　杜思远
学科专业　计算机应用技术
导　师　王昊
完成日期　2022 年 3 月

姓　　名：xxx　　学号：xxx

论文题目：基于深度学习的认知语言学转喻识别方法研究

上海大学

本论文经答辩委员会全体委员审查,确认符合上海大学硕士学位论文质量要求。

答辩委员会签名：

主任：

委员：

导　　师：

答辩日期：

姓　　名：xxx　　学号：xxx

论文题目：基于深度学习的认知语言学转喻识别方法研究

原 创 性 声 明

本人声明：所呈交的论文是本人在导师指导下进行的研究工作。除了文中特别加以标注和致谢的地方外，论文中不包含其他人已发表或撰写过的研究成果。参与同一工作的其他同志对本研究所做的任何贡献均已在论文中作了明确的说明并表示了谢意。

签　名：_____日 期：2022 年 3 月

本论文使用授权说明

本人完全了解上海大学有关保留、使用学位论文的规定，即：学校有权保留论文及送交论文复印件，允许论文被查阅和借阅；学校可以公布论文的全部或部分内容。

（**保密的论文在解密后应遵守此规定**）

签 名：_____导师签名：_____日期：2022 年 3 月

上海大学工学硕士学位论文

基于深度学习的认知语言学转喻识别方法研究

姓　名：XXX
导　师：XXX
学科专业：计算机应用技术

上海大学计算机工程与科学学院

2022 年 3 月

A Dissertation Submitted to Shanghai University for the

Degree of Master in Engineering

Detecting Metonymy in Cognitive Linguistics Using a Deep Learning Approach

MA Candidate：XXX

Supervisor：XXX

Major：Computer Application

School of Computer Engineering and Science,

Shanghai University

March 2022

摘　要

随着互联网信息技术蓬勃发展和国民经济水平快速提升，机器对文本智能处理的需求也逐渐增大。作为一种普遍的语言现象，转喻在日常的交流和书写中使用频率非常高。近些年来，深度学习方法发展十分迅速，在大多数自然语言处理任务上有明显的优势。但是，使用深度学习方法来识别转喻是否可行，目前仍缺乏相关研究与实验论证。因此，也就成为一个具有重要研究意义和应用价值的研究点。

从认知语言学的角度看来，识别转喻需要综合考虑实体本身和上下文信息，理解并构建实体词与上下文之间的交互，并找到实体词与上下文之间存在的语义冲突。然而，如何正确高效地表示实体的信息，以及如何充分地应用上下文中的句法和语义信息是转喻识别中的难点。因此，本文基于认知语言学中对于转喻研究的相关理论基础，开展基于深度学习的转喻识别方法研究，具体研究内容主要从词汇、句法、语义三个角度分别考虑如何提升模型对于转喻的识别性能：

（1）**融合上下文感知的实体词语义表示模型**。针对转喻文本中实体词汇含义无法精确表示的问题，本文提出了一种融合上下文感知的实体词语义表示模型。我们的模型以预训练语言模型 BERT 作为基线模型，通过实体指示器指示实体在句子中的语义和位置信息，增强实体词汇在转喻识别中的表达能力；同时联合训练句子级的向量表示和实体级的向量表示，清晰有效地表达上下文和实体的联合语义，提升了转喻识别的精确率和召回率。

（2）**基于注意力指导的图卷积网络句法约束模型**。针对文本中句式复杂、句法理解难的问题，本文提出了一种基于注意力指导的图卷积网络的软句法约束模型。我们将图卷积网络挖掘的依存句法信息融入特征向量表示中，利用基于注意力指导的依存关系权重分配方法，突出了句法关系中关键信息的表达；另外，深度融合句法和上下文语义表示，提升了模型在转喻识别任务中的准确率，实现了对文本中句法依存信息的有效利用。

（3）**基于变分信息瓶颈的语义压缩技术模型**。针对文本中信息冗余，有效语义特征难以抽取的问题，本文提出了一种基于变分信息瓶颈的语义压缩技术模型。通过学习从一个封闭的采样中生成中间数据表示，弥补了模型高度依赖训练数据的缺点，解决模型过拟合问题；而基于信息瓶颈的语义压缩过程去除转喻语句中的冗余信息，实现了语义知识的泛化和去噪，从而解决文本语义多样化问题，保证语义的精确性，有效提升了转喻识别的效果。

本文的研究有效地从认知语言学的角度使用深度学习方法解决了转喻识别问题，挖掘了转喻实体的真实含义，为知识图谱补全、机器翻译等下游任务提供了重要的技术支撑。

关键词：转喻识别、认知语言学、实体表示方法、软句法融入、信息瓶颈技术

ABSTRACT

With the rapid development of information on network, people's demand for language processing is gradually increasing. As a language phenomenon, metonymy is very common in the daily communication and writing. In recent years, the development of deep learning is rapid. In most natural language processing tasks, it has obvious advantages. Using deep learning to solve metonymy detection is significant and valuable.

From the perspective of cognitive linguistics, we need to consider both the entity itself and the context information, then construct the interaction between the entity and the context and find the semantic conflict. To efficiently represent the entity information and to fully apply the syntactic&semantic information are the biggest difficulties in metonymy detection. Therefore, this paper studies metonymy detection in cognitive linguistics using deep learning. The main content can be divided into vocabulary, syntax and semantics:

(1) **A model with context-aware entity representation**. To solve the problem that the meaning of entities in text cannot be accurately expressed, this paper proposes a context-aware entity representation model. Our model takes BERT as baseline and uses entity indicators to indicate the semantic and location information of entities, enhancing the expressive ability in metonymy detection. At the same time, the joint training of sentence and entity representation clearly expresses the joint semantics and improves the precision and recall of metonymy detection.

(2) **A syntactic constraint model using attention-guided graph convolutional networks**. To solve the complex sentence patterns and difficult syntactic comprehension, this paper proposes a soft syntactic constraint model using attention-guided graph convolution network. We incorporate the dependency syntactic information mined by graph convolution network into representation, then apply weight

assignment to highlight the expression of key information in dependency relations. In addition, syntactic and contextual representations are deeply integrated, which improves the accuracy of the model and makes effective use of dependency information in the text.

(3) **A semantic compression model using variational information bottleneck**. To solve the problem of information redundancy and extraction difficulty, this paper proposes a semantic compression model using variational information bottleneck. By learning to generate intermediate data from a closed sample region, the model makes up for the defect of highly dependent on training data and solves model overfitting. Moreover, the compression method using information bottleneck removes redundant information in sentences. It generalizes semantic knowledge and solves the problem of semantic diversity, which effectively improves the effect of metonymy detection.

In this paper, deep learning is used to effectively solve metonymy detection from the perspective of cognitive linguistics to mine the true meaning of metonymy entities. Finally, this method provides important support for downstream tasks such as knowledge graph completion and machine translation.

Keywords: Metonymy detection, cognitive linguistics, entity representation, soft syntax integration, information bottleneck

目　　录

摘　要.....VIII
ABSTRACT.....X
目　录.....XIII
第一章 绪论.....1
1.1　课题来源.....1
1.2　研究背景及意义.....1
1.3　研究问题.....3
1.4　研究内容.....5
1.5　研究的创新点.....7
1.6　论文的组织结构.....8
第二章 国内外研究现状.....11
2.1　相关概念.....11
2.1.1　转喻类型.....11
2.1.2　转喻和隐喻.....11
2.1.3　相关深度学习模型.....12
2.2　国内外研究概况.....13
第三章 融合上下文感知的实体词语义表示模型.....16
3.1　引言.....16
3.2　相关工作.....16
3.2.1　预训练技术.....16
3.2.2　语义表示方法.....18
3.3　问题定义和分析.....19
3.4　模型架构.....21
3.4.1　数据预处理.....22
3.4.2　转喻编码器.....22
3.4.3　转喻分类器.....23

3.5 实验验证 24
3.5.1 实验数据集 24
3.5.2 对比模型 25
3.5.3 实验结果与分析 26
3.6 本章小结 30
第四章 基于注意力指导的图卷积网络句法约束模型 31
4.1 引言 31
4.2 相关工作 31
4.2.1 图卷积神经网络 31
4.2.2 依存分析和融入 32
4.3 问题定义和分析 34
4.4 模型架构 36
4.4.1 表示编码层 36
4.4.2 表示对齐层 37
4.4.3 句法融入层 38
4.4.4 转喻分类器 41
4.5 实验验证 41
4.5.1 实验数据集 41
4.5.2 实验设置 42
4.5.3 对比模型 43
4.5.4 实验结果与分析 44
4.5.5 错误分析 53
4.6 本章小结 55
第五章 基于变分信息瓶颈的语义压缩技术模型 56
5.1 引言 56
5.2 相关工作 56
5.2.1 变分自编码器 56
5.2.2 变分信息瓶颈 58

5.3 问题定义和分析 59
5.4 模型架构 60
5.4.1 信息瓶颈压缩器 Compressor 61
5.4.2 变分正则化器 Regularizer 61
5.4.3 逻辑回归分类器 Classifier 62
5.5 实验验证 63
5.5.1 实验结果与分析 63
5.5.2 错误分析 68
5.6 本章小结 69
第六章 结论与展望 70
6.1 结论 70
6.2 展望 72
参考文献 74
作者在攻读硕士学位期间公开发表的论文 83
作者在攻读硕士学位期间所参与的项目 84
致 谢 85

第一章　绪论

1.1 课题来源

本课题来源于以下项目：

2021 年度上海市“科技创新行动计划”扬帆计划项目；项目名称：面向药物警戒的跨语言多知识驱动不良事件挖掘方法；项目编号：XXX。

1.2 研究背景及意义

近年来，各行各业都在不断地贡献文本数据资源，伴随互联网时代信息容量和技术的飞速提升，自然语言处理技术得到了迅猛发展。转喻现象在人类的语言环境中无处不在，在日常交际中使用频率很高。随着社交媒体的迅速崛起和文化交流的不断深入，转喻出现在更加多元化的平台上，受到更广泛的关注，而对于转喻的识别工作只是处于起步阶段，很多现有的方法无法为某些实体词找到准确的语义。

转喻识别这一任务旨在使计算机像人类一样分析和理解转喻现象，涉及计算机科学、认知语言学、神经科学等学科，具有极其重要的理论和实际应用价值【1】。作为一个自然语言处理上游基础任务，解决好转喻识别问题，就能准确理解文本中的实体含义，获得正确的语义信息，从而提高相关下游任务的准确性。比如，在信息检索【79】]【80】、机器翻译【81】【82】、阅读理解【21】**错误!未找到引用源。**、事件抽取【28】【46】和问答系统**错误!未找到引用源。** **错误!未找到引用源。**等领域，转喻识别的效果直接影响了整个应用系统的性能。

转喻在传统修辞学中被称为“借代”，指用凸显特征或相关事物指代某一事物【75】，是一种从源域到目标域之间的认知映射**错误!未找到引用源。**【74】。例如，在文本“马云收购饿了么”这句话中，“马云”是转喻的源域，马云的公司“阿里巴巴”是转喻的目标域，原句的实际意义为“阿里巴巴收购饿了么”。实现转喻的深层语义理解，可以将该知识融入到实体识别、关系抽取等任务中，提升这些任务的精度，帮助抽取出<**阿里巴巴**,收购,饿了么>这样的隐含知识，

可以在金融、医疗、物流等工业场景中得到应用。比如，当用户在搜索引擎中搜索“马云”时，扩展到“阿里巴巴”相关的内容，从而丰富内容的推送；同样地，通过挖掘出“阿里巴巴”和“饿了么”之间的隐含关系，可以实现企业知识图谱的补全；另外，在机器翻译系统中引入转喻识别的相关技术，可以在一定程度上提高翻译的准确率。

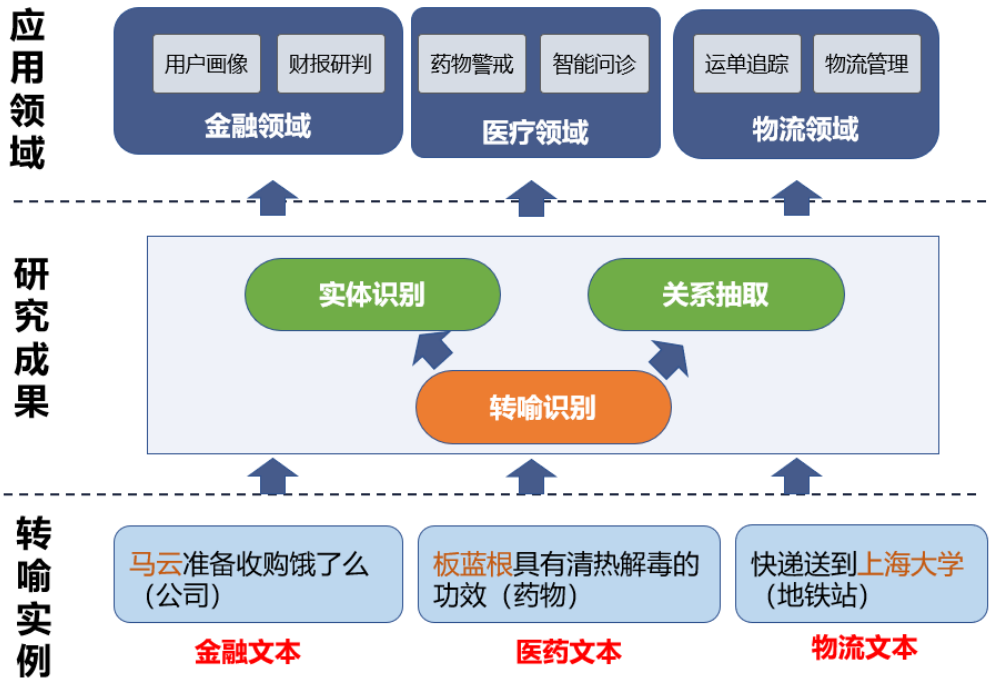

图 1.1 应用场景示例图

早年间，研究者通过计算语言学理解转喻问题。然而，基于计算语言学的方法依靠大量的概率计算来拟合模型，没有真正从认知规律上挖掘转喻信息。Lakoff【19】开启了认知语言学时代，对转喻识别进行了系统化的定义，在此之后，研究者们提出了大量转喻识别方法，从最初的支持向量机（SVM）**错误!未找到引用源。**机器学习统计模型，到长短期记忆网络（LSTM）【2】、卷积神经网络（CNN）【20】等网络结构来编码更深层次的上下文语义信息，再到近两年引入在大规模语料上预先训练的语言模型 ELMo **错误!未找到引用源。**、BERT【4】等，使得转喻识别的效果不断提升。近年来，陆续出现了基于深度学习方法，在转喻识别模型中占据了主导。不过，现有的基于深度学习的自然语言处理模型，受限于训练数据，自动机器推理和解释能力一般。

1.3 研究问题

Nissim 和 Markert【13】将转喻识别看作一个二分类任务，即识别实体在其上下文中是否具有转喻性。由于转喻词只能出现在特定的上下文中，脱离上下文的单一转喻词和非转喻词之间没有区别。因此，转喻与词汇本身的语义、句法结构、上下文信息是密不可分的。转喻识别通常伴随着两大理论，一种理论认为转喻实体通常与其所在上下文存在语义冲突，即句法上的词汇违规使用，该理论也被称作选择倾向性异常（Selectional Preference Violation），另一种理论认为转喻词在其特定上下文中所表达的语义与该词本身的常见语义存在冲突（Metaphor Identification Procedure）。转喻识别任务本质上是构建实体词与其上下文之间的交互，寻找两者之间的不同的互动方式。然而，Mao 等人【5】发现在转喻识别任务中，现有的端到端神经网络模型均采用通用的语义编码结构，没有显式利用认知语言学相关理论，造成关键知识的缺失。

基于上述认知，从认知语言学的角度出发，本文将转喻识别问题的关键点归结为实体信息增强和上下文知识的有效提取，需要说明的是，实体指转喻识别过程中需要判别的目标词，实体的真实语义通常不是实体在词典中预备的表面语义，会由特定的上下文环境的不同，展现出语义的变化和拓展。

我们在认知语言学的指导下，采集和处理转喻文本，对转喻识别中的实体表示及上下文环境语义理解进行了深入研究。其中，对上下文环境语义理解的研究分为句法融入和语义压缩两部分。具体研究问题如下：

（1）如何解决文本中实体语义难以精确表示的问题

语义理解依赖于完整的实体表示，从而提供快速精准的识别反馈。目前的深度学习模型相关研究没有为模型“指明”实体的各种显性信息，如实体语义信息，位置信息等，只有找到一种从转喻文本中准确获取实体信息并形成知识融入到模型的方法，才能为模型的准确识别提供保障。

大多数转喻识别的重要信息集中于实体的词嵌入表示中，但目前的转喻识别模型难以有效利用词嵌入表示知识，仅仅依靠原模型提供的上下文知识，准确率达到瓶颈。传统的依赖于人工特征提取的方法受到误差传播的影响，性能

不佳；近些年深度学习方法崛起，但这些模型的表示能力有限，同样性能受限；现有的预训练方法可以很好地表示上下文特征，但是依旧存在性能瓶颈，究其原因是没有充分利用文本中重要的实体特征。针对上述问题，需要对转喻文本中的实体表示问题进行研究，旨在充分利用实体特征，以增强实体词与上下文信息间的交互，从而提升转喻识别的准确率和召回率。

（2）如何解决依据句法结构理解复杂句式中的实体真实语义问题

一些转喻识别工作**错误!未找到引用源。**确定了转喻的理解离不开句法知识的辅助，从而给出正确的判断。转喻文本中有非常多句式复杂的长难句，例如“Engaged in very long range strategic bombing missions to enemy military, industrial and transportation, were Italy, France, Germany, Austria, Hungary, **Romania**, and Yugoslavia.（意大利、法国、德国、奥地利、匈牙利、罗马尼亚和南斯拉夫军队参与了对敌人的军事、工业和运输的远距离战略轰炸任务。）”。作为典型的兼含倒装与并列的句式，该句包含多层的逻辑嵌套。目前广泛采用的基于谓词（predicate）的转喻识别方法**错误!未找到引用源。**依赖于谓词窗口（predicate window）的选取，仅在窗口区域中加入句法信息，知识分布稀疏，不能满足全局句法特征融入的需求。综上所述，传统的序列模型**错误!未找到引用源。**无法提取有用的句法信息，而目前已有的句法模型**错误!未找到引用源。**，并没有真正地将词汇间的依赖关系转化为可用的句法知识，同样存在受句法解析器所产生的错误传播影响，设计一个高效的可以灵活融入句法信息的转喻模型迫在眉睫。

（3）如何解决文本上下文信息冗余且表述多样的问题

在转喻文本中，上下文信息冗余、表述多样的现象非常常见，造成语义理解的困难。转喻识别信息较为集聚，大部分有效信息都聚集于较小窗口区域（window）中，文本中常常包含大量冗余信息干扰模型的判断；另外，文本表述的多样性也可导致模型判断错误，如“马云收购饿了么”和“马云去饿了么”两个句式中，仅仅谓词的变化，实体“马云”就呈现出了不同的语义。

由于文本信息的冗余性和文本表述的多样性，现有的模型更侧重于通过高发生率的已知实例进行判断。假设目标实体在大多数转喻文本中未经过转喻，

而只以转喻的形式出现一次，因为数据的稀疏性，模型可能会过拟合，在下次预测目标实体时，更倾向于判断实体类型为非转喻。因此，解决语义信息冗余，降低语义稀疏性至关重要。

1.4 研究内容

转喻是一种句法和语义之间的违背关系【8】【9】，这种逻辑上的违背引起了认知语言学的广泛关注。转喻表达建立在源指称和目标指称之间稳固的关系之上**错误!未找到引用源。**【12】，当概念表示与实体在语义空间中相邻时，概念表示可能被激活，因此，目标实体的表面词汇语义是解决语义歧义的主要线索。

实体和句法都暗含转喻词的真实含义的信息，且句法【13】**错误!未找到引用源。**在转喻路径的构建中起着至关重要的作用**错误!未找到引用源。**，而依赖信息【15】【17】已被证明在关系抽取任务中帮助巨大**错误!未找到引用源。**；在最新的研究**错误!未找到引用源。**中，发现了信息压缩对语义去噪的突出贡献。基于上述认知，本文认为要解决转喻识别问题，应该同时考虑实体、句法以及语义三个层面。

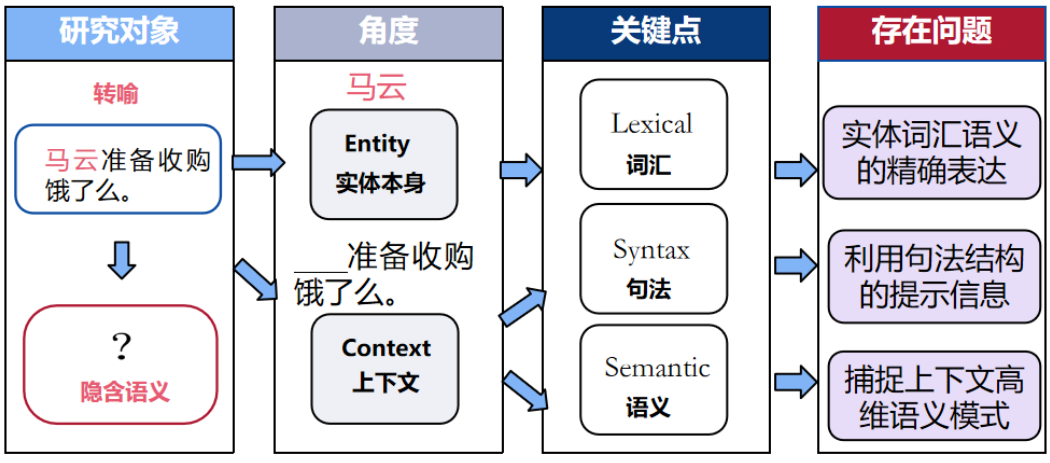

图 1.2 研究内容知识流图

针对上节提出的三个主要研究问题（见图 1.2），包含以下三个研究内容：

（1）为了解决文本中实体含义无法精确表示的问题，提出了一种融合上下文感知的实体词语义表示模型，提升实体表示精度。该模型充分利用实体特

征，增强实体信息对于判别结果的影响，高效组织转喻文本中的实体和上下文关系，为准确地表示实体和上下文的联合嵌入提供了坚实的基础。

在转喻文本中，关键信息大多集中在实体（entity）上。实体也被称为转喻任务的目标词（target word），其表示的完整性在很大程度上决定了模型的表现，充分利用实体词的语义、位置和结构信息能够有效地减少噪声的影响。

所以，针对转喻文本中实体信息难以提取和表示的问题，研究使用一种融合上下文的词汇语义表示方法，在上下文知识的联合指导下，有效提升实体语义表达能力，提升下游任务的识别精度和效率，同时深入研究实体表示的泛用性和准确性，解决实体词的分词表示差异问题；基于上述方法，可以进一步提出了一种融合上下文感知的实体词语义表示模型，即在具有上下文的环境中，实体的语义表达需要同时结合上下和其本身的语义，我们提出的模型使用基于BERT模型的神经网络联合训练实体和整个句子上下文，将句子级表示与实体词汇语义表示融合成更优秀的词嵌入向量表示，能有效利用实体中的关键信息，降低整个句子中噪声的影响，大幅提高了转喻识别的准确率和召回率。

（2）为了解决文本中句式复杂、句法理解难的问题，提出了一种基于注意力指导的图卷积网络句法约束模型，利用句法约束帮助实体语义重定向。认知语言学框架认为句法结构含有重要信息。例如，构造文法认定句法是语言的重要语义信息单元。现有的基于深度神经网络的转喻识别方法扫描整个句子序列的信息编码并压缩池化成向量表达，不能捕捉在自然语言信息传递中起到重要作用的句法结构；此外，以相同的权重将所有的句法依赖关系加入到句法表示向量中，无法区分各依存关系对于转喻识别的贡献。如何高效、准确地将句法结构约束引入到转喻识别中，是一个值得考虑的难点。

针对转喻文本中句式复杂、句法理解难的问题，本文提出了一种基于注意力指导的图卷积网络句法约束模型，通过基于注意力机制（Attention Mechanism）的图神经网络（Graph Convolutional Network，GCN），联合训练句子上下文语义和句法依赖的图向量表示。图神经网络高效地将句法依赖知识融入到特征向量的表示中，注意力机制则突出了依赖中关键信息的表达，在一定程度上解决了句法信息噪声的问题。

（3）为了解决文本中信息冗余，有效语义特征难以抽取的问题，提出了一种基于变分信息瓶颈的语义压缩技术模型，提升上下文去噪和泛化能力。在转喻文本中，文本的语义冗余和表达的多样性造成了语义信息稀疏和缺失，阻碍了句子语义的进一步理解。研究利用信息瓶颈策略，解决上下文语言表述多样性以及信息冗余问题，实现对语义的准确识别。

针对文本中信息冗余和语义稀疏的问题，提出了一种混合变分信息瓶颈框架。该框架主要分为以下两个模块：**1. 上下文正则化变分编码器：**基于变分推断的数学公式和已知模型，学习了一种跨域的信息映射技术，消除词汇偏差，挖掘了隐藏语义，弥补了模型高度依赖训练数据的缺点，增强了模型抗干扰能力。**2. 实体信息瓶颈压缩器：**通过朴素信息瓶颈的结构，选择性地聚合上下文的实体信息，实现了语义泛化和去噪。基于以上框架，聚合提炼实体语义向量和上下文语义向量，有效消除了噪声的影响，捕捉了固定的高维语义模式。

通过以上研究，实现了对转喻识别语义的深度理解，有效的提升转喻识别的识别精度。

1.5 研究的创新点

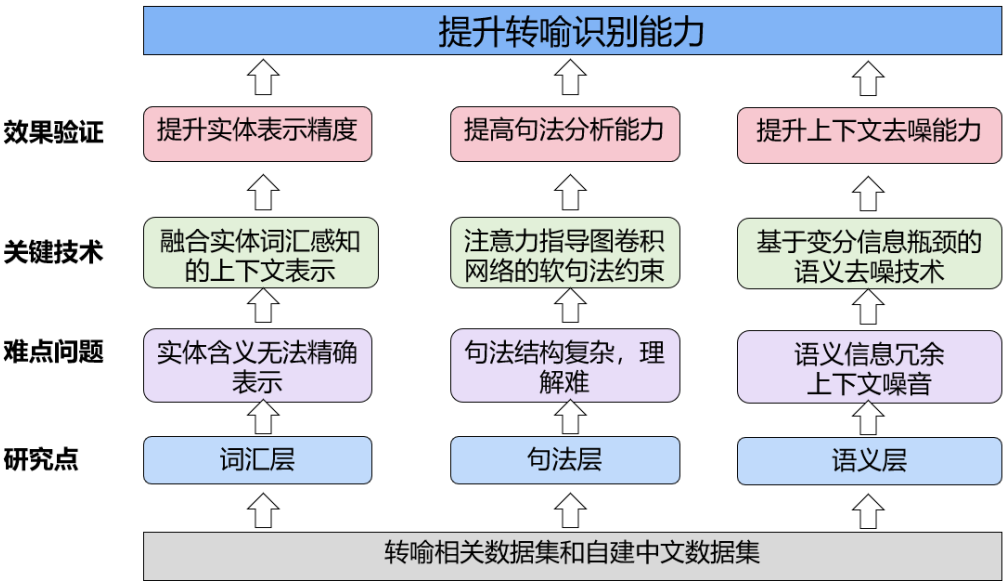

图 1.3 研究具体创新点展示

该工作的主要创新点是基于认知语言学理论，提出了基于深度学习的转喻识别方法，从词汇、句法、语义三个不同层面，对实体、上下文、句法在转喻识别中起到的作用进行了研究，具体创新点如下：

（1）针对文本中实体含义无法精确表示的问题，提出了一种先进的联合感知方法，清晰地感知实体语义，并准确地融入词向量表示中；开发了一种融合上下文感知的实体词语义表示模型，不仅高效地组织了实体和上下文的知识结构，而且充分利用了实体词汇的信息，减弱了词汇的多样表达带来的影响，为有效地融入实体知识提供了坚实的保障，印证了实体词汇语义的重要性。

（2）针对文本中句式复杂、句法理解难的问题，提出了一种自动分配权重的句法融入方法，免除了手动增删依赖关系的麻烦，提高了依赖关系权重的识别准确率；构建了一种基于注意力指导的图卷积网络句法约束模型，该模型将句法知识融入到特征向量的表示中，突出了转喻文本中关键句法信息的表达，同时，对不同的句法依赖关系自动赋予合适的权重，不仅提升了转喻的准确率和召回率，而且赋予了句法知识融入可解释性，证明了句法信息在转喻识别任务中的重要性。

（3）针对文本中信息冗余及上下文语言表述多样，有效语义特征难以抽取的问题，提出了一种混合变分信息瓶颈框架，减少了噪声的影响，增强了对于多样化数据的鲁棒性；构造了一种基于变分信息瓶颈的语义压缩技术模型，不仅将词向量的表示融入到了正则化器中，减少了语义信息的冗余，而且学习从一个封闭的采样中生成中间数据表示区域，弥补了模型高度依赖训练数据的缺点，实现了对转喻语义的深度理解，为下一步稀疏化的转喻模式提取奠定了技术基础。

。

1.6 论文的组织结构

本文首先介绍了基于深度学习的认知语言学转喻识别方法的研究背景和意义，然后针对转喻识别过程中的词汇、句法、语义三个环节进行了深入研究，最后对本论文的研究工作进行了总结。本文的组织结构如下：

第一章介绍了认知语言学转喻识别方法的研究背景和研究意义，针对转喻文本数据的特点，提出了三个主要的研究问题及研究内容，最后总结了本论文的创新点。

第二章介绍了转喻识别的国内外研究概况及相关技术，包括国内外的重要研究成果以及本文相关的神经网络结构等。

第三章提出了一种新颖的联合感知方法和一种融合上下文感知的实体词语义表示模型。首先对转喻识别中实体的表示方法进行介绍和分析；然后具体阐述了联合感知方法和融合上下文感知的实体词语义表示模型；最后，通过实验验证了所提出的转喻识别方法和模型的有效性。

第四章提出了一种自动分配权重的句法融入方法和一种基于注意力指导的图卷积网络句法约束模型。首先对转喻识别句法融入的问题和困难进行介绍和分析；然后具体阐述了自动分配权重的融入方法和软句法约束模型；最终通过实验验证了所提出的句法融入方法和模型的有效性。

第五章提出了一种端到端的混合变分信息瓶颈框架和一种基于变分信息瓶颈的语义压缩技术模型。首先对文本语义解析存在的问题进行了介绍和分析；然后具体阐述了混合变分信息瓶颈框架与基于变分信息瓶颈的语义压缩技术模型；最后，通过实验验证了所提出的方法和模型的有效性。

第六章对本文的研究工作进行了总结，并提出了几个有待进一步研究的方向。

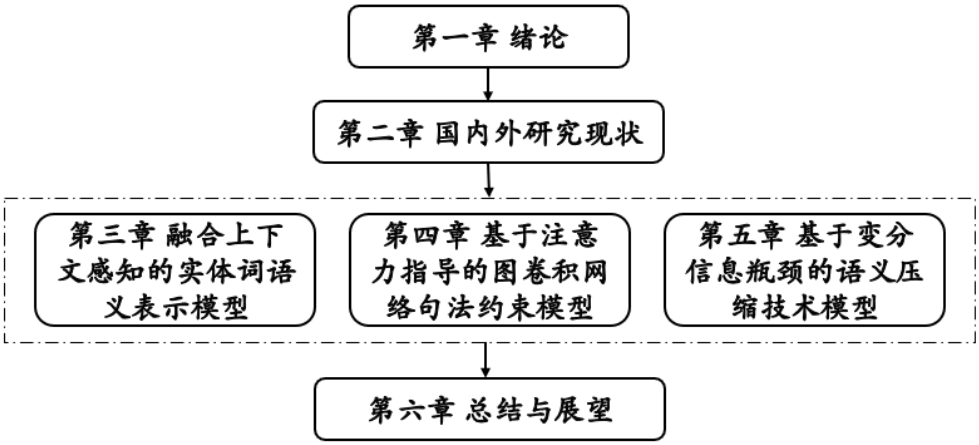
第一章 绪论
第二章 国内外研究现状
第三章 融合上下文感知的实体词语义表示模型
第四章 基于注意力指导的图卷积网络句法约束模型
第五章 基于变分信息瓶颈的语义压缩技术模型
第六章 总结与展望

图 1.4 论文组织结构

第二章　国内外研究现状

2.1 相关概念

转喻识别是自然语言处理领域中一个新兴且具有重要意义的研究方向, 本文从词汇、句法、语义三个方面研究基于深度学习的认知语言学转喻识别方法。本文在研究过程中运用了多种相关技术和概念，以下作详细的阐述。

2.1.1 转喻类型

依据转喻实体的不同，可以分为多种类型。以下列举几种常见的转喻类。

（1）**地点转喻**：地点转喻指转喻实体为一个地点，是一种常见的转喻类型，也是本文主要的研究类型。例如: “**华盛顿**对**北京**目前的态度非常不满。”“华盛顿”指代的是“美国”，进一步可以引申为“美国政府”，而“北京”则可以被转喻为“中国”，进一步可以引申为“中国政府”。

（2）**人物转喻**：人物转喻指转喻实体为一个人物。例如: “**马云**收购饿了么。”其收购的实体其实并非“马云”本人，而是“马云”的公司“阿里巴巴”。

（3）**组织转喻**：组织转喻指转喻实体为一个组织。例如: “使用**文华财经**进行基金交易。”“文华财经”指代的是一个手机 app 交易软件。

2.1.2 转喻和隐喻

转喻是一种修辞手段，当甲事物同乙事物不相类似，但有密切关系时，可以利用这种关系，以乙事物的名称取代甲事物。转喻的重点不在于“相似”，而是在“联想”，转喻现象大多出现在相关实体上。

隐喻又称暗喻，是指用一种事物暗喻另一种事物，即把未知的东西变换成已知的术语进行传播的方式，例如，“轿车**甲虫**般地前行”这个隐喻就假定，我们不知道轿车怎么运动，但知道甲虫匆匆穿过地面的行进模样，即把甲虫的特征变换成了轿车的特征。 除了在名词实体上，动词上也会出现隐喻现象。

转喻和隐喻有明显的区别，具体表现为：

（1）映射基础不同：从认知语言学的视角看，隐喻建立在相似（similarity）之上，转喻则建立在相关（relevance）或者邻近（adjacency）原则之上。隐喻和相似之间存在一种辩证关系，语言中既有基于相似性的隐喻，同时隐喻也可以创造相似性。

（2）映射模式不同：隐喻是不同概念域之间的映射，是系统结构投射，由两个域构成，一个是结构相对清晰的源域（source domain），另一个是结构相对模糊的目标域（target domain）。理想认知模型（ICM）是指特定的文化背景中人对某领域的经验和知识做出的抽象的、统一的、理想化的理解，是建立在认知模型上的一种复杂的、整合的完型结构。Lakoff【18】认为隐喻是基于从一个理想认知模型投向另一个理想认知模型的映射，通过映射限定了两个 ICM 之间的关系。一般而言，隐喻的映射是单向的，而转喻则是双向进行的。

转喻和隐喻是两种不同的认知现象，在国内隐喻的研究开始得较早也较多，研究内容也相对完善。由于中文中的转喻并不明显，且大多数情况下研究者将转喻看作是隐喻的一种特殊形式，因此，对于转喻的研究较少。 随着互联网时代语言的发展，对于语义认知的要求越来越高，亟需更新的研究方法来突破目前的认知瓶颈，提升机器对于本文的认知理解能力。

2.1.3 相关深度学习模型

（1）自编码器

作为第五章中变分自编码器的基础构造，自编码器应用广泛，能够通过无监督学习，学到高效表示的人工神经网络。自编码器可作为强大的特征检测器（feature detectors）应用于深度神经网络的预训练，此外，还可以随机生成与训练数据类似的数据，形成生成模型（generative model），如用人脸图片训练自编码器以生成新的图片。和主成分分析（PCA）类似，自编码器能进行数据压缩（data compression），从原始数据提取最重要的特征。假设经过多层的训练，自编码器学习到一个良好的特征来表示原输入数据，可以在自动编码器的最顶层添加一个分类器，如 LR 回归、支持向量机 SVM **错误!未找到引用源。**等，利用

梯度下降方法对整个网络进行有监督的微调，完成了这个有监督训练后，该神经网络便可以高效完成分类任务。

（2）卷积神经网络

卷积网络**错误!未找到引用源。**（Convolutional Neural Networks，CNN）是一个很好的计算机科学借鉴神经科学的例子，也是第一个解决重要商业应用的神经网络，在诸多应用领域都表现优异，其最大的特点是在多个空间位置上共享参数。卷积运算是一种数学计算，和矩阵相乘不同，卷积运算可以实现稀疏相乘和参数共享，压缩输入端的维度。另外，卷积神经网络不需要和循环神经网络一样为每一个神经元所对应的每一个输入数据提供单独的权重，与池化（pooling）相结合后，卷积网络可以被理解为一种公共特征的提取过程。卷积神经网络通常由输入层、卷积层、池化层、全连接层构成，在第三章中，我们使用 CNN 模型在中文转喻数据集上实验，并和我们提出的模型进行对比。

（3）循环神经网络

循环神经网络（Recurrent Neural Networks，RNN）是一类用于处理序列数据的神经网络，能以高效率对序列的非线性特征进行学习。循环神经网络依靠前一时间步积累的信息进行预测的能力可以应付相关信息与待预测词距离较小的情况。 但是，如果是长距离间隔的信息，循环神经网络往往会因为“梯度消失”的问题而无法进行记忆。长短期记忆网络（LSTM）【2】是一种特殊的循环神经网络，它通过输入门、遗忘门和输出门三个“门”函数来控制输入值、记忆值和输出值，这种设计不仅能很好地捕捉时间序列的语义，还解决了循环神经网络结构中的“梯度消失”问题，对间隔距离较大的信息有更好的处理能力。很多耳熟能详的模型，如 BiLSTM 等，都是在 RNN 的基础上进行改进得到，本文中，多次使用 RNN 模型进行模型对比。

2.2 国内外研究概况

转喻是日常生活中常见的语言现象，涵盖多个领域【18】**错误!未找到引用源。**，转喻识别被定义为一项旨在识别文本中的转喻实体的任务，已有的转喻识别方法主要分为基于特征和基于神经网络的方法。

大多数基于特征的方法在 SemEval 2007 Shared Task 8 基准测试集中测试及完善【23】。例如，Nissim 和 Markert【13】基于潜在的转喻词抽取特征；Farkas 等人【24】在提取的特征集上使用最大熵分类器改进识别效果；Brun 等人**错误!未找到引用源。**利用无监督的方法计算句法上下文在大型数据集上的相似性；Nastase 和 Strube **错误!未找到引用源。**扩展了 Markert 和 Nissim【23】的特征集，且使用了更复杂的特征，比如 WordNet 3.0 以及 WikiNet（维基百科的分类网络）；Nastase 等人**错误!未找到引用源。**在概率框架下研究了局部和全局的上下文语境；Nastase 和 Strube【27】使用了支持向量机，并建立了以维基百科为基础的大规模知识库。类似方法在相关任务中有一定的发展前景，然而，由于特征提取过程高度依赖人工，普遍存在误差传播的问题，并且构造特征集也需要外部的 NLP 工具和额外的预处理成本。

如今，大多数转喻识别工作都使用深度神经网络。Melamud 等人【42】使用长短期记忆（LSTM）的语义特征，从大型语料库中学习通用上下文词嵌入；Gritta 等人**错误!未找到引用源。**提出了一种基于长短期记忆网络架构的谓词窗口模型 PreWin，通过 LSTM 拥有了较强的编码能力。在编码过程中，PreWin 只保留谓词周围的单词和其对应的依赖标签，以排除嘈杂的依赖信息；Mathews 和 Strube【28】利用 BERT 初始化了单词的嵌入向量，而不是微调后将其作为分类器使用；Li 等人**错误!未找到引用源。**提出了一种词遮蔽方法，在转喻数据集上大大优于现有方法。词遮蔽方法用[X]标记替换了目标实体表示，在不考虑任何实体信息的情况下，迫使模型只根据上下文做出预测。

最近，预训练技术在许多 NLP 任务中取得了巨大的成功。预训练语言模型预先在大规模标注数据集上训练，然后在下游任务上微调，产生优秀的上下文表示，不需要从头开始学习参数。得益于强大的表示能力，预训练模型在捕捉上下文或结构特征方面表现突出【31】，在许多 NLP 任务中显著超过了其他颇有竞争力的神经网络模型【30】**错误!未找到引用源。**。因此，将预训练模型引入到转喻识别任务中是一个自然而然的做法。

与国外相比，国内转喻研究主要集中在语言学方向，计算语言学中的相关研究较少，由于转喻和隐喻任务的强烈相关性，本文主要介绍隐喻的国内研究

现状。赵红艳**错误!未找到引用源。**利用最大熵模型和条件随机场，并添加了同义词信息和语义信息，实现了隐喻识别；贾玉祥等人【33】基于词语抽象度的隐喻识别方法，通过跨语言迁移技术得到词语抽象度。实验表明，词语间的抽象度在隐喻识别任务中十分有效；游维和周昌乐【34】通过统计大规模语料库及向量空间模型，构建了隐喻句生成系统；苏畅和周昌乐**错误!未找到引用源。**在隐喻理论的基础上，使用了知识发现和语料库相结合的方法，对隐喻进行理解。

综上所述，需要对于转喻进行更为深入的研究。

第三章　融合上下文感知的实体词语义表示模型

3.1 引言

转喻识别任务的本质是构建实体与上下文之间的交互，探明并解决实体词与上下文之间的语义冲突，所以，实体词和上下文表示的精确性和结构性十分关键。传统转喻模型只使用静态的词向量语义表示来完成分类，无法准确地表达转喻识别中的重要冲突，转喻识别的准确率不高。因此本章拟解决以下两个子问题：1）如何增强实体词汇信息在转喻识别任务中的语义表达准确性；2）如何构建一种融合上下文感知的实体词语义表示模型，使得二者的联合表示可以清晰有效地表达对应语义。

3.2 相关工作

3.2.1 预训练技术

转喻识别作为文本领域的重要研究课题，其识别的准确性高度依赖于模型框架的语义理解能力和信息挖掘能力。Transformer 编码器是由 Google **错误!未找到引用源。**发布的一款最早用于机器翻译的神经网络框架。虽然循环神经网络有一定的记忆功能，但是循环结构训练难度过大，且训练时间过长。Transformer 框架创造性地抛弃了循环神经网络的做法，改用注意力机制对传统的 Encoder-Decoder 框架进行了改进，帮助框架更多地关注范围内的语境信息，避免句子过长导致的网络退化。另外，Transformer 可以方便地利用现有的主流设备并行工作，大大提升了运算效率。

Transformer 框架主要由 Encoder 和 Decoder 构成。Encoder 由 6 个相同的层组成，每个相同的层都包含两个子层（sublayer），即多头注意力层 (multi-head attention layer) 和全连接层（fully connected feed-forward network），且在各层输出时，添加了残差连接和层归一化操作。具体表示为：

$$SublayerOutput = LayerNorm\big(x + SubLayer(x)\big) \tag{3.1}$$

在子层中，Transformer 使用了多头注意力机制（multi-head attention），通过 h 个不同的线性变换对 Q，K，V 投影，并将多种不同的 Attention 映射拼接起来：

$$MultiHead(Q，K，V) = Concat（head_1，\ ...head_n）W^o \quad (3.2)$$

$$head_i = Attention(QW_i^Q, KW_i^K, VW_i^V) \quad (3.3)$$

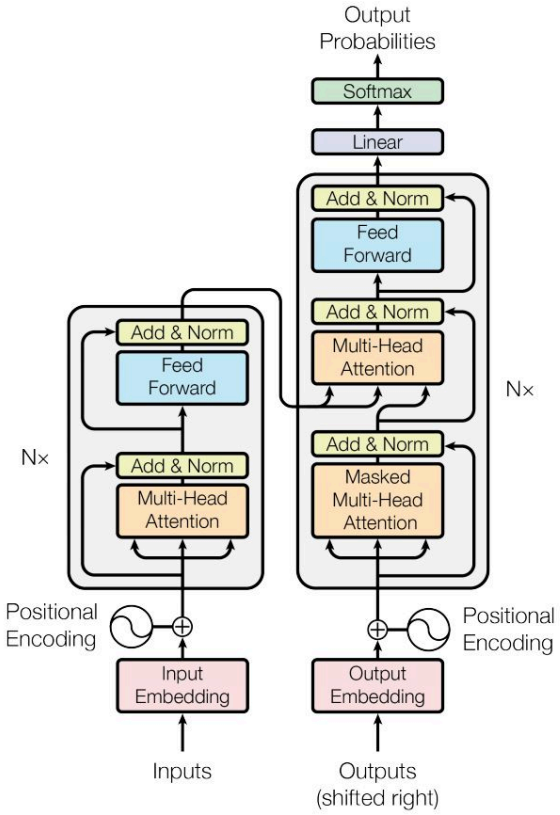

图 3.1 Transformer 结构 **错误!未找到引用源。**

Attention 采用了 scaled dot-product 计算，公式表示为：

$$Attention(Q，K，V) = softmax（\frac{QK^T}{\sqrt{d_k}}V）\quad (3.4)$$

Decoder 的结构和 Encoder 比较相似，但是相比于 Encoder 增加了一个注意力的子层。由于 Transformer 相较于 RNN 和 CNN 等结构有明显的优势，Transformer 成为了 BERT 和 XLNet **错误!未找到引用源。**等大型预训练语言模型所采用的基础组件。

预训练的思想最早在图像理解领域萌芽，此后渐渐发展到自然语言处理领域，预训练词向量是预训练在自然语言处理领域最常见的一种应用。很多自然语言处理任务中的标注语料十分有限，不足以训练出足够好的词向量，通常使

用与当前任务无关的大规模未标注语料进行词向量的预训练，目前，很多深度学习模型都倾向于使用经过预训练的词向量（如 Word2Vec【41】和 GloVe **错误!未找到引用源。**等）进行初始化，加快网络的收敛速度。

预训练词向量只为每个词赋予对应的词向量，而不考虑其上下文的信息，例如，要对“apple”设置相应词向量，“apple”可能有水果的含义，也可能表示某一具体的电子产品公司，而不管上下文语境如何变化，对它设置的词向量却完全一致，导致实体语义不明。所以，为了在设置词向量时考虑上下文的信息，发展了 Context2Vec **错误!未找到引用源。**、ELMo **错误!未找到引用源。**等模型，取得了不错的效果。BERT 是一个直接在深层 Transformer 网络上进行训练的语言模型，通过预训练和微调在很多自然语言处理下游任务中取得了最好的结果【76】**错误!未找到引用源。**【78】。不同于其他深度学习模型，BERT 在训练前联合调整各层次的上下文，得到关于每个词的深度双向表示，经过训练的 BERT 模型通过特定任务的输出层进行微调，在很大程度上解决了一词多义问题。相较于循环神经网络，依赖于 Transformer 模型训练的 BERT 能更有效地捕获长距离依赖关系，并且准确理解当前上下文每个 token 的语义。

3.2.2 语义表示方法

转喻识别领域最早使用一系列基于特征工程的方法表示语义。Nissim 和 Markert【13】最先使用句法修饰关系和同义词表对决策分类器进行训练评估，以克服数据稀疏性和泛化的问题；Markert 和 Nissim **错误!未找到引用源。**对 2003 年发布的特征集进行了更新，增加包括：潜在转喻词（PMW）的标签（如 subc、obj）、转喻词的修饰语、转喻词的限定词、转喻词的标签、转喻词的单词数量以及当前上下文中转喻词的语法角色数量特征；Farkas 等人【24】的 winning 系统利用上述特征和最大熵分类器实现了较高准确率，该系统是当时最轻便精简的系统，但仍然额外构建了大量特征，并挂载了一些外部工具，花费大量时间和人力成本；Brun 等人【26】使用经过改良的 Xerox 深度解析器生成局部句法和全局分布特征，它是唯一一种通过在大型语料库 BNC（英国国家语料库）上计算得出句法上下文相似度的无监督方法；Nastase 和 Strube **错误!未**

找到引用源。使用了支持向量机 (SVM) 和人工提取的特征，包括从 BNC 中提取的语法特征、WordNet 3.0、维基百科的类别网络等等。庞大的特征库使模型达到了很高的准确率，但是额外的训练和特征、外部工具的过度使用问题依然无法解决；Nastase 和 Strube 【27】拓展了此前 2012 年的工作**错误!未找到引用源。**，将 Wikipedia 转换为大规模多语言概念网络 WikiNet，通过建立在 Wikipedia 上的类别网络，自动发现新的关系及实体。转喻识别任务作为他们的评价任务之一进行了测试评估，该模型获得了迄今为止基于特征的模型的最高性能。

此后转喻识别进入了深度学习时代，语义的表示开始不局限于多样化的特征集，而是使用在大规模数据集上训练得到的预训练词向量。有研究者使用 GloVe**错误!未找到引用源。**预训练词向量和 BiLSTM**错误!未找到引用源。**模型从左到右（Left-to-Right）和从右到左（Right-to-Left）分别编码每个单词的上下文信息，得到了不错的效果。类似地，ELMo[83]也是基于大量文本训练深层 BiLSTM 网络结构的语言模型，通过在词向量的学习中考虑深层网络不同层的信息，并加入到单词 token 的最终表示中，取得了实质性的提升。Gritta 等人**错误!未找到引用源。**提出了一种基于 BiLSTM 网络架构的谓词窗口模型 PreWin，在上下文表示之外，额外加入了 one-hot 向量形式的句法知识表示。Li 等人**错误!未找到引用源。**在 2020 年使用 BERT 预训练模型解决转喻识别，得益于大规模预训练知识和数据增强方法，其性能上大大优于其他的模型。

3.3 问题定义和分析

图 3.2 实体与上下文关系展示图

目前转喻识别的研究关注于如何使用单一向量更高效地表示转喻文本的上下文语义，比如 Li 等人**错误!未找到引用源。**，使用预训练语言模型达到了较高的性能，然而，此类模型无法清楚表示实体词汇语义，造成实体信息不完整，

限制了模型准确率的提升。实体语义无法清晰表示的原因，主要可以归结为两点：

（1）**实体词在转喻文本中没有被清晰地表达**。如图 3.2 中所示，“马云”作为转喻实体，承担重要的指示作用。最新的转喻识别方法**错误!未找到引用源。错误!未找到引用源。**仅仅把转喻实体随转喻文本一起输入到模型中，没有对转喻实体进行任何额外的信息增强，导致实体的语义信息、位置信息缺失，使得转喻识别率不高。

（2）**缺少合适的实体词汇语义和上下文语义联合表示方法**。上下文和转喻实体词共同诠释了句子的含义，表达了两者之间的语义冲突，起到了联合判断的作用，二者缺一不可。目前的转喻模型研究**错误!未找到引用源。**，只关注如何提高上下文表示的质量，而几乎没有研究者开展关于如何表达二者联合语义的研究。如图 3.2 所示，“马云”本身作为人物的语义与“收购饿了么”的上下文语义产生了冲突，从而判断“马云”经过转喻。所以，如何表达两种语义间的关联关系，以及如何构建一种基于大规模预训练语言模型的实体上下文联合表示方法是重大的难点。

因此，本章首先就大规模预训练模型中的语义表示方法进行研究。其识别方法整体设计如图 3.3 所示，过程如下：

①实体词汇信息提取：给定一个标注了目标实体的句子，我们在句子中加入实体指示器，并提取出实体的语义信息；

②模型输入：将句子和实体词汇序列输入基于 BERT 的转喻识别模型；

③转喻识别：基于 BERT 的转喻识别模型利用实体词汇信息和句子的上下文语义信息生成一个关系向量表示，并进行转喻识别预测；

④模型输出：根据模型的预测结果预测出目标实体的转喻类型。

综上，本章的贡献在于：

- 以大规模预训练语言模型 BERT 作为语义理解基础，形成较强的上下文语义表示能力。
- 加入实体指示器，指示实体在句子中的语义和位置信息，增强实体词汇信息在转喻识别任务中的表达能力。

- 设计语义融合层，构建一种融合上下文感知的实体词语义表示结构，形成具有语义感知的实体上下文联合表示，使这种表示可以清晰有效地表达对应语义。

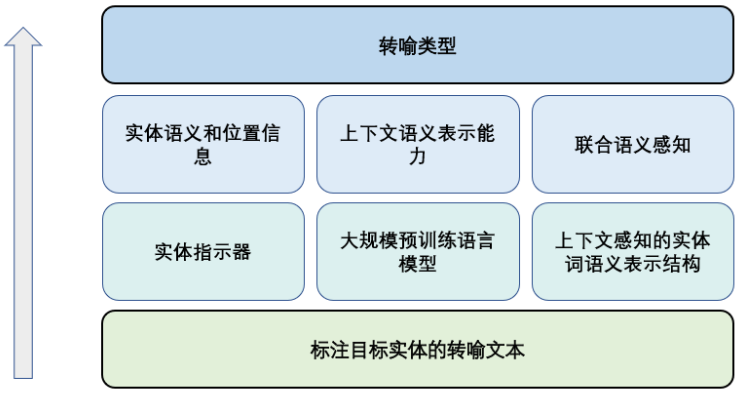

图 3.3 词汇转喻识别方法整体设计

3.4 模型架构

为了充分利用实体词汇的信息，减弱文本中无关词汇的影响，本文提出了融合上下文感知的实体词语义表示模型 EBERT。该模型的整体结构主要分为四层，从下往上分别是输入、BERT 编码器、转喻分类器和输出，如图 3.4 所示。

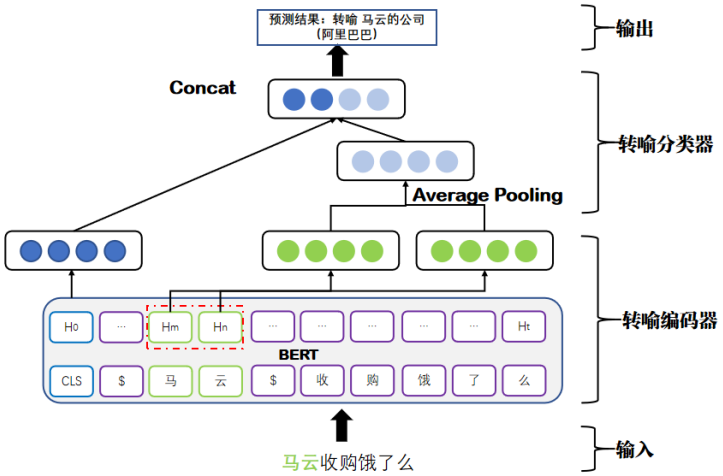

图 3.4 融合上下文感知的实体词语义表示模型 EBERT

3.4.1 数据预处理

大规模预训练模型 BERT 的输入表示方法是固定的，为了适应不同的任务需求，开发了单个句子表示和一对句子表示两种输入方式，其中，单个 token（中文文本中，输入时切分每个字为一个单独的 token）的输入表示由对应的词汇嵌入、段嵌入和位置嵌入构成。依据 BERT 预训练模型的初始设置，标识符“[CLS]”被附加到每个序列的开头，作为序列的第一个 token，而“[CLS]”的最终隐藏状态被用作分类任务的句子上下文表示，如果在一个任务中有一对句子，则使用标识符“[SEP]”分隔这两个句子。

EBERT 的输入由两部分组成，分别是已标记实体的句子和对应的实体词汇信息。对于已标记实体的句子，模型在实体的开头和结尾插入实体指示器，以帮助模型捕获实体的位置信息并增强实体词汇信息在转喻识别任务中的表达能力。例如，转喻文本“2018 年，[**马云**]$_{e1}$ 收购饿了么。”中，“[**马云**]$_{e1}$”是目标实体，我们尝试在实体前后加上不同的或者相同的实体指示器，第一种方案为在实体之前插入指示器“$”，在后一个实体之前插入指示器“#”，第二种方案为在实体之前和之后同时插入“$”，经过实验比较后，选择第二种方案应用到模型中，处理后的输入形式为“2018 年，$**马云**$收购饿了么。”。

3.4.2 转喻编码器

初始的句子文本 S 与词汇信息表示 E 一起输入模型后，通过基于 BERT 的转喻编码器得到转喻向量表示。

（1）上下文语义表示

在 BERT 原始实现中，BERT 模型将特殊标识符“[CLS]”添加到 token 序列的首位置。“[CLS]”在 BERT 所对应的隐藏状态向量，对应于整句句子的所有 token 向量表示经过了全连接层聚合而成的表示，我们提取该表示作为此句的上下文向量表示，称为 H_0。

（2）实体语义表示

对于一个实体，可能有许多 token 组成，比如实体“马云”，由 token“马”和“云”构成。设 H_x 是转喻文本句子中第 x 个 token 的向量表示，m 是目标实体 e_1 在句子中的开始位置，n 是目标实体 e_1 在句子中的结束位置，此时，可以用 H_m 和 H_n 之间的向量序列表示实体 e_1。EBERT 通过平均池化（Average Pooling）获得目标实体的单一向量表示 H_e，如公式(3.5)所示：

$$H_e = \frac{1}{n-m+1}\sum_{t=m}^{n} H_t \tag{3.5}$$

（3）实体与上下文表示聚合

上文中阐述了 BERT 的两种输入表示设计，分别为单个句子表示和一对句子表示。在执行文本相似度或问答任务时，需要同时输入一对句子（相似度测算时为测算语句 1 和测算语句 2，问答任务为问题和回答）来进行表示。在转喻识别任务中，目前所有的研究都使用传统的单个句子表示，受到关系抽取任务中使用一对句子表示方法表示【47】的启发，我们将实体词汇信息作为第二个句子的输入送入 BERT 模型，具体来说，在句子表示 H_0 之后，加入“[SEP]”分隔符进行分割，并在之后连接实体表示向量 H_e。

为了使 BERT 的表示更加鲁棒，在拼接上下文表示、分隔符、实体表示后，连续添加两个全连接层，权重参数分别为 W^* 和 W'，偏置参数分别为 b^* 和 b'，得到当前实例文本的最终向量表示 H_{final}，如下式所示：

$$H_{final} = \rho(W^*[\rho(W'\text{concat}[H_0; H_e] + b']) + b^*]) \tag{3.6}$$

3.4.3 转喻分类器

在通过基于 BERT 的转喻编码器获得当前实例的关系向量表示 H_{final} 之后，运用一个全连接的 softmax 层生成所有预定义关系类型的概率分布，并施加 dropout 函数解决可能出现的过拟合问题，如公式(3.7)所示：

$$p(y|x,\theta) = softmax(H_{final}) \tag{3.7}$$

其中，$y \in Y$ 是句子中的正确转喻类型，θ 指神经网络中所有可学习的参数，包括权重参数 W^* 和 W' 等。

3.5 实验验证

3.5.1 实验数据集

为了评估模型的性能，本文在多个公开的英文转喻数据集和自建的中文转喻数据集上进行实验验证。

（1）SemEval 英文基准转喻数据集

SemEval【23】是一个地点转喻数据集，由英国国家语料库中的数据构建而成，包括 925 个训练数据和 908 个测试数据。SemEval 数据集中的类别分布约为 80%非转喻类（literal，比如地理上的领土和政治实体）、18%转喻类（metonymy，比如发生在某地的事件或者某地的人）和 2%混合分类（mixed，即无法区分是否是转喻），以模拟日常生活中转喻类与非转喻类的自然分布比例。

（2）ReLocaR 英文转喻数据集

ReLocaR **错误!未找到引用源。**使用维基百科 Random Article API 中的样本构建而成，它包含 1,026 个训练样本和 1,000 个测试样本，ReLocaR 中非转喻类和转喻类的分布比例约为 5:5。该数据集相比于 SemEval 数据集有以下几个优点：

- 去除了转喻难以区分的下级标签，仅仅分为转喻和非转喻两大类。
- 转喻和非转喻数据的标签分布更均衡。
- 该数据集由经过训练的标注者标注，保证了标注的质量。

（3）CoNLL 英文转喻数据集

CoNLL【28】包含了大约 6,215 个样本，该数据集由命名实体识别任务 CoNLL 2003 Shared Task 修改得到，但全程只由一个标注者标注，文本存在一定的噪音，其转喻类的样本是非转喻类样本的两倍，且平均的样本句子长度大于 SemEval 和 ReLocaR 数据集。

（4）自建中文转喻数据集

由于目前转喻识别任务处于起步阶段，没有成熟的中文数据集可供测试使用，我们利用英文转喻数据集，通过文本翻译、人工调整及筛选和人工标记步骤构建了自建中文转喻数据集。

①文本翻译：以 SemEval、ReLocaR、CoNLL 英文数据集作为源数据集，使用接口翻译源数据集中的转喻文本，最终得到中文转喻样本。

②人工调整及筛选：考虑到接口翻译得到的样本数据集质量较差，我们舍弃所有不符合中文表达规范的转喻文本并对合格的句子进行人工校正。

③人工标记：我们对转喻文本中的目标实体进行人工标注，以获得实体的信息。

经过以上步骤，得到一系列带有实体标记的中文转喻样本，本文利用这些文本数据验证我们的工作在中文文本上的有效性。最终，该数据集包含 1,986 个带有实体标记的样本，并随机划分了其中的 1,192 个样本作为训练集，794 个样本作为测试集。

3.5.2 对比模型

fastText 模型：fastText 是 Facebook 于 2016 年开源的一个词向量计算和文本分类工具。fastText 在学术上并没有太大创新，但是优点非常明显，在文本分类任务中，fastText 能取得和深度网络相媲美的精度，却在训练时间上比深度网络快许多数量级。

CNN 模型：面相于文本分类的经典 CNN 模型，由输入层，卷积层，池化层和 softmax 层组成。本模型在 CNN 的传统图像模型架构上做了一些调整以适应文本数据。

BiLSTM+Att 模型：在普通的 BiLSTM 上增加了注意力层，大大提升了其表示能力。

Paragraph, Immediate 和 PreWin 模型：Paragraph、Immediate 和 PreWin 模型**错误!未找到引用源。**都是以深度学习模型 BiLSTM 作为基线模型。它们可以同时将 token 编码成语义向量【39】【44】**错误!未找到引用源。错误!未找到引用源。**，并将依赖关系标签编码成独热（one-hot）向量。这三种模型在选择 token 的方式上有所不同，Immediate x 直接选择实体的左侧和右侧的 x 个单词作为模型的输入；Paragraph 模型扩展了 Immediate 模型的功能，从每个实体的一

侧取 50 个单词作为分类器的输入；PreWin 提出了谓词窗口的概念，只选择谓词周围的单词作为输入。

PreWin（BERT）模型：该模型为使用了 BERT 嵌入的 PreWin 模型。模型没有将 BERT 作为分类器，而是将原始的 GloVe 嵌入替换为了 BERT 嵌入来初始化其单词向量。

BERT-MASK 模型：该模型发布于 2020 年，是转喻识别领域最新的研究成果。该模型在 BERT 的基础上使用目标词遮蔽的方法微调，在许多数据集上证明了相比于 BERT 模型的优越性。

3.5.3 实验结果与分析

（1）自建中文转喻数据集的词汇模型实验结果

表 3.1 自建中文转喻数据集的词汇模型实验结果

模型	Acc	Precision	Recall	F1-L	F1-M
fastText	70.0	70.4	70.1	71.6	68.8
CNN	73.1	73.2	73.1	73.3	73.3
BiLSTM+Att	73.1	73.3	73.4	73.9	72.4
BERT	81.7	81.6	81.6	75.4	87.7
EBERT	**85.3**	**85.1**	**85.0**	**78.4**	**91.9**

表 3.1 展示了本文提出的融合上下文感知的实体词语义表示模型以及四种常用的神经网络模型在自建中文转喻数据集上的实验结果，其中，四种转喻识别对比模型分别是 fastText、CNN、BiLSTM+Att 和 BERT。本文通过 5 个评价指标对 5 个模型在转喻识别任务中的性能进行了全面评估，依次是准确率（Acc）、精确率（Precision）、召回率（Recall）、非转喻类的 F1 值（F1-L）、转喻类的 F1 值（F1-M）。从实验结果可以看出，fastText、CNN 和 BiLSTM+Att 在中文转喻识别任务上差距不大，BERT 模型凭借预训练模型的强大表示能力大幅提高了指标。EBERT 在各项评价指标上都达到最佳，相比于 BERT， EBERT 在各项指标上平均高出约 4 个百分点，显示了巨大的性能优势。

（2）英文数据集的词汇模型实验结果

为了进一步验证融合上下文感知的实体词语义表示模型的有效性，我们在SemEval、ReLocaR 和 CoNLL 英文数据集上进行了多组对比实验。

表 3.2 展示了我们的模型和多种先进的深度学习神经网络模型在 CoNLL、ReLocaR、SemEval 数据集上的表现。预训练模型 BERT 在转喻数据集上获得的最佳效果超过了基于 BiLSTM 的系列模型；而我们也在预训练模型 BERT 上施加了谓词窗口方法，创建了 PreWin（BERT）模型，该模型和 BERT 相比，取得了不小的提升；Li 等人**错误!未找到引用源。**的 BERT-MASK 模型，使用了目标词遮蔽的方法，弥补了 BERT 的短板，取得了非常不错的结果，相比之前的方法有比较大的提升；而本文提出的融合上下文感知的实体词语义表示模型 EBERT 相比于 BERT-MASK，其性能更佳，在三个数据集上准确率分别达到 94.7%，95.5%和 89.2%，优于目前最先进的模型。

表 3.2 英文数据集的词汇模型实验结果

模型	CoNLL（Acc）	ReLocaR（Acc）	SemEval（Acc）
Paragraph	-	80.0	81.3
Immediate-5	-	81.4	81.3
Immediate-10	-	81.3	81.9
PreWin-GloVe	87.9	83.6	83.1
BERT	89.5	91.3	84.7
PreWin（BERT）	92.6	92.2	87.1
BERT-MASK	93.9	94.4	88.2
EBERT	**94.7**	**95.5**	**89.2**

为了进一步证明方法的效果，我们设置了实验进行额外的验证。

（3）词汇模块消融实验

本实验进一步挖掘除 BERT 模块以外的其他模块的贡献。为此，我们创建了三个对比模型：

1. **EBERT-NO-SEP-NO-ENT**：在 EBERT 的基础上，丢弃在实体两侧的指示器以及丢弃了实体语义向量，也就是仅仅使用标识符“[CLS]”对应的向量来进行分类。

2. **EBERT-NO-SEP**：在 EBERT 的基础上，丢弃在实体两侧的指示器但保留了实体和上下文语义向量的联合表示。
3. **EBERT-NO-ENT**：在 EBERT 的基础上，丢弃实体语义向量但保留了实体两侧的指示器。

观察表 3.3 可知，以上三种模型都比 EBERT 表现更加糟糕。其中 BERT-NO-SEP-NO-ENT 表现最差，证明了实体指示器和联合表示方法都对模型做出了重要贡献。

在转喻识别中，关系标签依赖于上下文和目标实体的语义。模型如果没有对实体做特殊的指示，则无法定位目标实体，丢失关键的实体信息，而实体指示器可以将实体位置信息传递到 BERT 预训练模型当中。另一方面，上下文和目标实体向量的聚合进一步丰富了表示蕴含的语义信息，帮助模型获得更高的准确度。

表 3.3 消融实验准确率对比

模型	CoNLL（acc）	ReLocaR（acc）	SemEval（acc）
EBERT-NO-SEP-NO-ENT	89.5	91.3	84.7
EBERT-NO-ENT	94.4	94.5	86.4
EBERT-NO-SEP	94.3	94.6	87.0
EBERT	**94.7**	**95.5**	**89.2**

（4）实体信息贡献实验

为了进一步研究实体信息在转喻任务上的突出贡献，我们额外进行了实体信息的贡献测试。

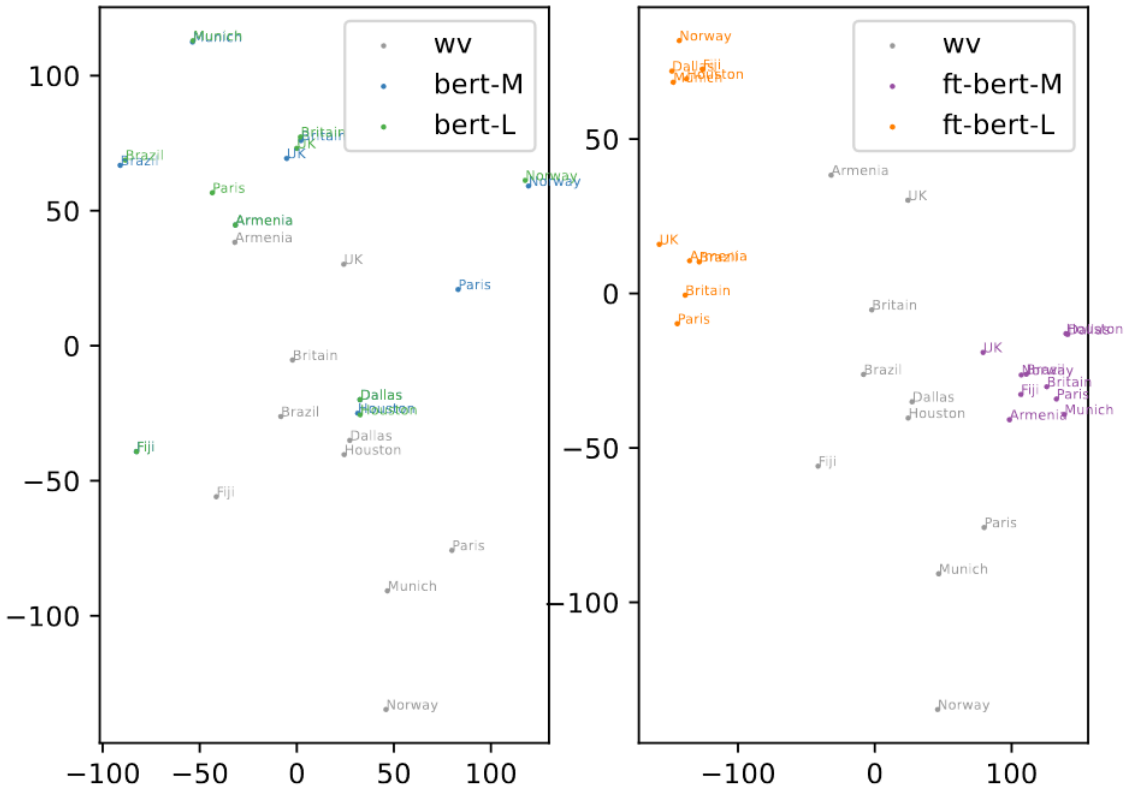

图 3.5 实体信息的贡献对比

该实验不使用上下文表示，仅仅把单一的实体表示输入到 BERT 模型中，图 3.5 展示了 BERT 在微调前后的语义分布图对比（左图为微调前，右图为微调后）。其中，“wv”指实体的原始语义表示分布，“bert-M”指转喻实体在微调前的 BERT 中的语义表示分布，“bert-L” 指非转喻实体在微调前的 BERT 中的语义表示分布，“ft-bert-M”指转喻实体在微调后的 BERT 中的语义表示分布，“ft-bert-L” 指非转喻实体在微调后的 BERT 中的语义表示分布。

左图中显示，在 BERT 微调前输入实体表示，其语义表示的分布与原始语义相差不大，表明了在 BERT 模型没有微调前，实体信息表示的能力非常弱，而右图中显示，在 BERT 微调后，转喻和非转喻的实体被分成了两个簇，表明了微调后的模型拥有了区分转喻和非转喻实体的能力。

现有的面向转喻识别的深度学习模型仅仅依靠上下文表示进行判断。上下文表示的形成方法是将句子中的每个 token 输入全连接层，得到的上下文表示必然含有很多噪音，而通过实体表示贡献的实验，证明了实体表示输入到大规模

预训练语言模型中可以大大提高转喻的识别准确率，验证了模型设计思路的正确性。

3.6 本章小结

本章针对转喻文本中的实体表示问题进行了研究。针对转喻文本中实体含义难以精确表示的问题，提出了基于大规模预训练语言知识的联合感知方法，使模型输出的语义表示处于实体和上下文知识的联合指导下，解决了单一上下文表示方法的实体信息丢失问题。针对转喻文本噪声多，实体信息难以得到充分利用的问题，提出了一种融合上下文感知的实体词语义表示模型 EBERT，采用基于 BERT 的神经网络对上下文和实体语义进行信息聚合，形成更优秀的语义向量表示。通过这种方式，模型充分利用实体词特征，增强了实体词与上下文信息间的交互，从而提升转喻识别的准确率和召回率。实验验证了实体信息在转喻识别任务中的重要性，并证明了本章所提出的模型能够有效地解决转喻识别中实体信息的缺失。

第四章　基于注意力指导的图卷积网络句法约束模型

4.1 引言

转喻识别通过实体与上下文的关系推理，将实体表面含义与指代事物联系起来，是自然语言处理领域中一个极具挑战性的课题。近年来，基于 BERT 的模型取得了不错的效果，然而，此类模型不是基于局部知识，如目标实体【48】，就是基于全局知识，如上下文**错误!未找到引用源。**，对于距离实体较远的信息和知识，会因为引入额外噪声而导致融入效果变差。因此，我们认为转喻识别过程应既依赖于词汇语义，也依赖于句法知识。然而，已有的句法知识融入方法对所有依赖“一视同仁”，文本中蕴含的某些无用的句法信息极大干扰了转喻的识别准确率。所以，本章拟解决以下两个子问题：1）如何解【50】决转喻文本中长难句识别困难的问题；2）如何构建一种句法依赖权重分配方法，使得句法知识中的噪声可以被有效地清除。

4.2 相关工作

4.2.1 图卷积神经网络

深度学习在图像处理以及自然语言处理两个领域都取得了巨大的成功，作为深度学习的代表模型，卷积神经网络能解决很多问题，但是它研究的对象仅仅限制在欧几里得域的数据，该域最显著的特征就是有规则的空间结构，比如图片是规则的正方形栅格，语音是规则的一维序列。

卷积神经网络可以高效地处理一维或二维的矩阵表示的数据结构，而很多数据并不具备规则的空间结构，比如推荐系统、电子交易、计算几何、脑信号、分子结构等抽象出的图谱等，这些图结构的每个节点连接都不尽相同，具有不规则的数据结构，在这种情况下，图卷积神经网络（Graph Convolutional Network）应运而生。图卷积神经网络是一种能对图数据进行深度学习的方法，它的核心思想是利用边的信息对节点信息进行聚合从而生成新的节点表示，并

同时对节点特征信息与结构信息进行端对端学习，适用于任意拓扑结构的节点与图，在节点分类与边预测等任务上，其效果远远优于其他方法。

4.2.2 依存分析和融入

依存句法分析（ Dependency Parsing，DP）通过分析语言单位内成分之间的依存关系揭示其句法结构。相较于使用语义刻画句子，依存分析的好处在于不需理解词汇本身的含义，而是通过词汇所承受的语义框架来描述该词汇，而其数目相对词汇来说数量小很多，于是，大部分的句子都可以用这个框架来表示。

我们从成分间的支配关系来阐述依存分析的原理。首先，句子中的核心动词是支配其他成分的中心成分，它本身不受支配；其次，其它成分间也存在支配关系。关于如何支配的问题，具体可以总结为以下五条规律**错误!未找到引用源。**：

（1）一个句子中只有一个成分是独立的，即核心成分；

（2）句子的其他成分都从属于某一成分，即除了核心成分外的部分；

表 4.1 部分依存关系标签类型

关系类型	Tag	Description	Example
主谓关系	SBV	subject-verb	我送她一束花 (我 <-- 送)
动宾关系	VOB	直接宾语，verb-object	我送她一束花 (送 --> 花)
间宾关系	IOB	间接宾语，indirect-object	我送她一束花(送 <-- 她)
前置宾语	FOB	前置宾语，Fronting-object	他什么书都读(书 <-- 读)
兼语	DBL	Double	他请我吃饭(请 –> 我)
定中关系	ATT	Attribute	红苹果(红 <-- 苹果)
核心关系	HED	Head	指整个句子的核心

（3）任何一个成分都不能依存于两个及以上的成分；

（4）如果成分 A 直接从属成分 B，而成分 C 在句子中位于 A 和 B 之间，那么，成分 C 或者从属于 A，或者从属于 B，或者从属于 A 和 B 之间的某一成分；

（5）核心成分左右两边的其他成分相互不发生关系，相当于核心成分是一条界线，左右两边的部分不再发生支配关系。

表 4.1 展示了常用的几种依存关系类型，运用这些关系类型可以进行依存分析。依存关系对文本的理解和识别至关重要，例如，有两句相似问句： ①马云的儿子是谁？②马云是谁的儿子？使用古老的词袋模型判别句子的语义，两句句子会得到相同的结果，这显然不合理，额外考虑其句法结构是必要的。

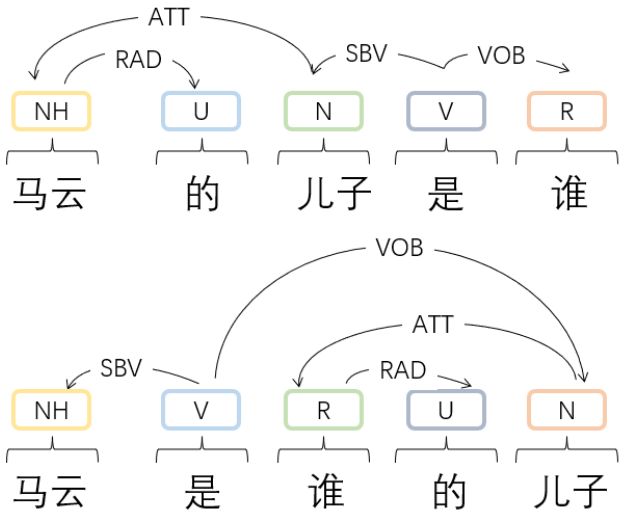

图 4.1 依存关系重要性分析

如图 4.1 所示，对两句句子实施依存分析后，句子间句法依存树的不同帮助模型理解了真实的句子语义。

最近的研究频繁地在各种自然语言处理任务中利用依赖关系【49】【50】【53】【54】。虽然依赖集成方面的相关研究显示了很不错的研究前景，但是目前为止，在转喻识别上的进展依然有限。受近期其他 NLP 分类任务【17】的启发，我们深入研究了依赖融入在转喻识别任务上的有效性。

在数据预处理阶段，已经有大量的工作将依赖关系作为特征进行集成。Kambhatla **错误!未找到引用源。**训练了一个统计分类器，通过结合来自文本的不同词汇、句法和语义特征提取依赖关系；Zhang 等人**错误!未找到引用源。**研究了在解析树中嵌入句法结构特征的方法，但这类方法普遍对语言结构的变化很敏感。最近的研究倾向于使用基于图的模型来集成深度神经网络和依赖解析树，研究者们提出了多种依赖于预定义规则的硬剪枝策略以提取有用的依赖信

息，例如，Xu 等人**错误!未找到引用源。**使用了依赖树中实体之间最短的依赖路径构建依赖关系；Liu 等人【57】分别使用递归神经网络和卷积神经网络合并了目标实体和附属于最短路径的子树之间的最短依赖路径；Miwa 和 Bansal【58】沿着实体的解析树或最近公共祖先（LCA）下的子树执行自底向上或自顶向下的计算来充分利用依赖信息；Zhang 等人【16】删除了两个目标实体之间可能存在的关系的最短路径附近的单词来减小噪声影响。这些硬剪枝方法虽然能有效去除不相关的依存关系，但同时也可能会剔除一些有用的信息，为了解决上述冲突，Guo 等人【17】提出了一种软剪枝模型 AGGCN，该模型通过 GCN 将原始的依赖树转换为完全连接的边加权图，使有用的依赖关系能够得到重视，并剔除无用的依赖关系。目前的依赖关系融入的研究仅仅在关系抽取任务上开展【16】【17】，将句法约束纳入到基于 BERT 的转喻识别模型中具有非常重要的意义。

4.3 问题定义和分析

转喻被视为一种句法语义上的违背，这种逻辑违背引起了认知语言学届广泛的研究兴趣。先前的研究**错误!未找到引用源。**【12】揭示了转喻是建立在源指和目标指之间的实际且坚实的关系基础上的，在语义空间中，如果概念外延与原实体相邻，则可能激活概念外延，因此，目标实体的表面含义是解决语义歧义的主要线索。研究【13】认为，句法暗含大量转喻信息，在转喻依赖构建中起着至关重要的作用。某些依赖关系，如“某人在**西雅图**动物园”中的修饰关系“西雅图-动物园”，可以帮助判别“西雅图”为非转喻实体，而“**西班牙**赢得世界杯”中的主谓关系“西班牙-赢得”，可以帮助判别“西班牙”为转喻实体。以往的实验表明，依赖信息能够提升关系抽取任务的准确率【17】，因此，我们认为在转喻识别任务中应充分考虑句法信息。给定一个句子，我们可以通过使用一些现成的 NLP 工具包进行依存分析来获得两个词之间的依赖关系。图 4.2 展示了依赖关系帮助模型进行转喻识别的一个例子，“马云”与“收购”之间的 nsubj 关系表明了“马云”和“马云的公司阿里巴巴”之间强烈的指代关系，而“2018 年”与“收购”之间的 nmod 依赖关系贡献较小，然而，以往的方法

将两种关系等同对待，并设置相同的权重，导致对“马云”一词的准确含义产生误解。根据中文语法中的语言配价，上述句子中转喻实体“马云”的语义主要由动词词根“收购”决定，因此，有必要开发一个高效利用句法知识的转喻识别模型。

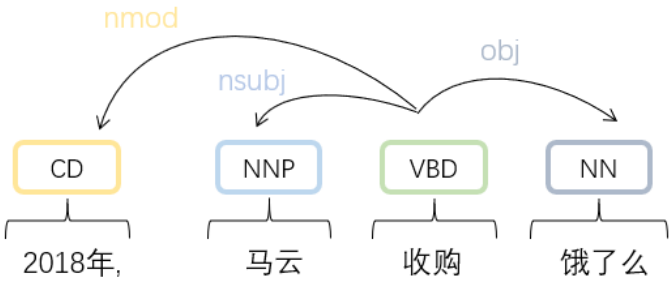

图 4.2 转喻文本依存分析举例

本章首先就基于注意力指导图卷积网络的图卷积网络句法约束模型进行研究，其识别方法整体设计如图 4.3 所示：

①依存分析：给定一个标注了目标实体的转喻样本，使用已有的 nlp 工具对样本进行依存分析，获得任意两个 token 之间的依存关系，构建依存关系图；

②模型输入：将句子的上下文语义信息和句法信息输入基于 BERT 的转喻识别模型；

③转喻识别：Jawahar 等人【59】提出，BERT 在捕获复杂的语法模式和距离较远的依赖信息方面存在一定困难。为了捕获更多的长距离依赖关系，我们提出了一个基于注意力的图模型，将依赖关系和 BERT 表示合并到图卷积网络中。与之前的工作有很大的区别，我们的模型既没有直接根据依赖解析树构造神经网络，也没有将依赖特征连接到输入嵌入中，而是使用基于注意力指导的图卷积网络来施加软依赖约束。最终，我们的模型有效地利用了高权重的依赖关系，并去除了不相关的句法噪声。

④模型输出：根据模型的最终表示预测出目标实体的转喻类型。

我们工作的贡献在于：

- 不同于前人直接使用 one-hot 编码的方式，我们使用图卷积网络融入依存句法信息，高效且完整地融入句法知识。
- 使用多头注意力机制为句法依赖实行自动权重分配，解决了句法信息的噪

声问题，提高识别效率。

- 使用深度连接方法为句法和语义表示实行深度信息融合，增强了最终表示的表达能力。

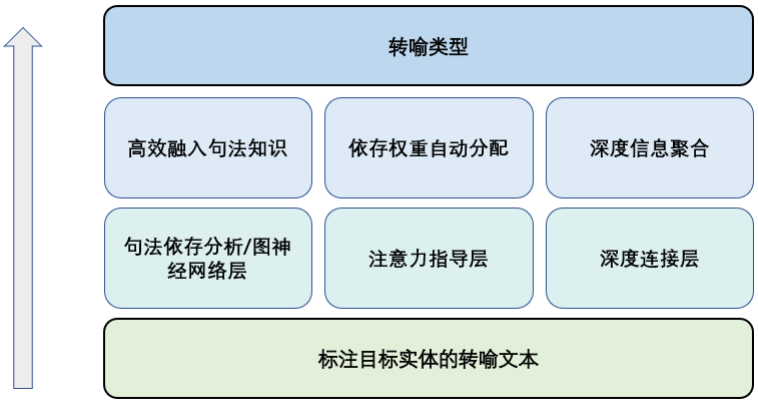

图 4.3 句法转喻识别方法整体设计

4.4 模型架构

为了解决长难句中转喻识别率低和句法权重分配效率差等问题，本章提出一种新颖的基于注意力指导的图卷积网络句法约束模型EBERT+AGCN，通过引入句法依存树结构知识来丰富原始语义信息。首先，使用 EBERT 编码器编码带有实体的转喻样本，并通过对齐层依次输出每个单词的深度表示，继而对句子进行依存关系分析，提取出依赖关系图并转换为邻接矩阵，和 BERT 表示一起提供给句法融入层。然后，使用句法融入层输出的整合句法知识和语义知识的表示预测转喻标签。该模型的整体结构主要可以分为四层，从下往上依次为表示编码层、表示对齐层、句法融入层和转喻分类器，如图 4.4 所示。

4.4.1 表示编码层

第三章介绍了在转喻任务中 EBERT 模型相比于 BERT 模型的优越性，在本章中，模型直接使用 EBERT 代替 BERT 进行编码，进而产生深层的双向表示。EBERT 表示编码层将输入打包为*[CLS, St, SEP]*，其中 *CLS* 是表示句子开始的标识符，*St* 是 WordPiece Tokenizer（BERT 分词器）生成的 token 序列，*SEP*

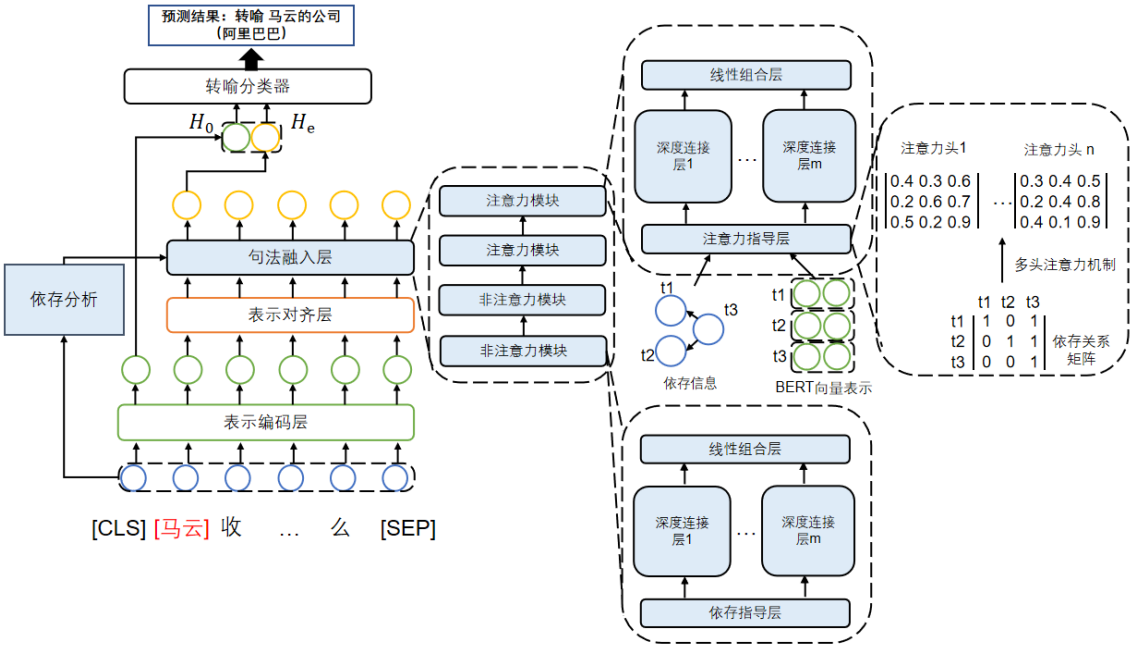

图 4.4 基于注意力指导的图卷积网络句法约束模型 EBERT+AGCN

是表示句子结束的标识符。模型将打包的句子 S 作为输入，并计算上下文表示，对于每个对应于句子第 x 个 token 位置的表示h_x^0，分别连接原始 token 的实体嵌入S_x^{tok}，位置嵌入S_x^{pos}，段嵌入S_x^{seg}：

$$h_x^0 = concat\left[S_x^{tok}; S_x^{pos}; S_x^{seg}\right]. \tag{4.1}$$

在通过 N 个连续的 transformer 编码器的模块后，编码器得到了第 x 个位置的编码表示h_x^N：

$$h_x^N = Transformers(h_x^0) \tag{4.2}$$

4.4.2 表示对齐层

在表示对齐层中，BERT 使用 WordPiece Tokenizer 进一步将单词分割成 WordPiece 列表，例如，把单词“played”分割为[play， ##ed]。然而，依存分析不会执行这一步的分割，只对每一个完整的单词进行依赖的分配，此时产生了分词上的冲突，因此，我们将输出 token 与原始单词对齐，并通过对 token 表示采用平均池操作来重构每个单词表示。假设 h_x，…， h_y 为 WordPiece 的 BERT 表示，x、y 分别表示单词 piece 列表的开始索引和结束索引，得到整个原始单词的嵌入为：

$$\widetilde{h_i} = \frac{1}{y-x+1}\sum_{t=x}^{y} h_t \tag{4.3}$$

设 $H = [h_1, \cdots, h_n]$为一个转喻语句，除了上述对齐方法，还可以构造一个映射矩阵 M，将 WordPiece 的列表表示 H 转换为完整的单词表示。投影矩阵 M 记录了从原始单词到 WordPiece 的转换，并作为一个转换器来恢复逐词的表示：

$$\widetilde{H} = H\boldsymbol{M}^{\boldsymbol{T}} \tag{4.4}$$

上式中，M 是一个投影矩阵，其中$\boldsymbol{M} \in R^{m\times n}$，$m$ 表示输入句子的单词个数，n 表示用 WordPiece 表示之后的新长度。

4.4.3 句法融入层

句法融入层通过添加额外的句法信息完善了最终的表示。我们效仿 Guo 等人【17】的方法，在句法融合层中设计了两类句法整合模块，即非注意模块和注意模块，各模块的总体架构展示在图 4.4 的右侧。

分别设计注意模块和非注意模块可以更便利地处理数据。非注意模块显式地融入依存分析图，而注意模块则自动学习图中边的权值。句法融入层将前一个模块输出的对齐表示作为输入，并将依赖关系信息融入了对齐表示中，使其能同时感知上下文和句法特征。

注意模块在结构上与非注意模块相似，但注意模块的注意力矩阵的初始化是通过自注意力机制生成的，而非注意模块的注意力矩阵直接使用依赖邻接矩阵 A 生成。A 的选择取决于使用的是 GCN 架构或者 AGCN 架构，表示为:

$$\tilde{\tilde{A}}\begin{cases} A, & if\ non-attention \\ \varphi\left(\widetilde{H}\times\widetilde{H}^{T}\right), if\ attention \end{cases} \tag{4.5}$$

φ是一个注意力函数，如 additive **错误!未找到引用源。**、general dot-product **错误!未找到引用源。**或 scaled dot-product **错误!未找到引用源。**，经过对比，最终模型使用了 scaled dot-product。

非注意模块由依存指导层、深度连接层、线性组合层组成，注意模块由注意力指导层、深度连接层、线性组合层组成，下面我们对每个子层作介绍。

（1）依存指导层

该层通过图卷积操作将每棵依赖树转换成相应的邻接矩阵 A，如果 token i 和 j 之间存在依赖关系，则 $A_{ij}=1$，否则 $A_{ij}=0$。在多层图卷积网络中，通过对从第 1 到第 $l-1$ 层进行图卷积运算，得到 token 表示$\tilde{h}_i^{(l)}$：

$$\tilde{h}_i^{(l)} = \rho\left(\sum_{j=1}^{n} A_{ij} W^{(l)} \tilde{h}_j^{(l-1)} + b^{(l)}\right) \tag{4.6}$$

其中$W^{(l)}$表示权重矩阵，$b^{(l)}$表示偏置向量，ρ是激活函数。$\tilde{h}_i^{(l)}$表示当前层的隐藏状态。在每个图的卷积过程中，每个节点会收集并压缩图中相邻节点的信息。

（2）注意力指导层

现有的方法采用依赖关系来施加硬句法约束，即用 1 表明依存关系存在，0 表示依存关系不存在，如果想要去除句法约束中的噪声，就需要基于专家经验的手工规则，在浪费大量人力成本的同时，效果并不显著。

作为手工规则的改进方案，注意力指导层采用了软修剪（soft-pruning）策略去除句法噪声。在这一层中，注意力模块通过多头注意力**错误!未找到引用源。**生成权值范围为 0～1 的软邻接矩阵$\tilde{A}^{(t)}$，$\tilde{A}^{(t)}$由以下公式得到：

$$\tilde{A}^{(t)} = softmax\left(\frac{QW_i^Q \times (KW_i^K)^T}{\sqrt{d}}\right) \tag{4.7}$$

Q 和 K 分别是多头注意力的 query 和 key，Q 和 K 都等于前一层输入的表示 $\tilde{H}$，d 表示$\tilde{H}$的维度，$W_i^Q \in R^{\mathrm{d}\times\mathrm{d}}$和$W_i^K \in R^{\mathrm{d}\times\mathrm{d}}$ 都是可学习的参数，$A^{(t)}$是对应于第 t 个注意力头的邻接矩阵。通过以上过程，将由 0 和 1 组成的矩阵 A 替换为了软注意力矩阵$A^{(t)}$，句子中的依赖关系，特别是间接的、多跳的关系，由多头注意力机制得到梳理和明确。传统的剪枝方法往往去除了被认为是不相关的依赖关系，丢失关键的信息，而软剪枝方法为每个依赖关系分配其应有的权重，有效避免了信息丢失的问题。

（3）深度连接层

之前的工作**错误!未找到引用源。**已经证明，深度连接的图卷积网络有助于捕获结构信息，于是，我们创建深度连接层来学习更多的全局信息，并训练更

深层的 GCN 模型。每个深度连接层都有 L 个子层，这些子层按顺序依次放置，每个子层都以前面所有子层的输出作为输入，如图 4.5 所示。首先计算由初始表示以及前面每一子层产生的表示串联而成的表示$g_j^{(l)}$：

$$g_j^{(l)} = \left[\tilde{h}_j^{(0)}, \tilde{h}_j^{(1)}, \ldots, \tilde{h}_j^{(l-1)}\right] \tag{4.8}$$

其中$\tilde{h}_j^{(0)}$是由对齐层输出的初始表示，$\tilde{h}_j^{(1)}, \ldots, \tilde{h}_j^{(l-1)}$是前面每一子层产生的表示。为提高参数效率，对子层中的表示维度 d_{hidden} 进行了缩减，用公式表示为 $d_{hidden} = d/L$，其中 L 为子层数，d 为输入维数，例如，设子层数 L 为 3，输入的维度为 768，则 $d_{hidden} = d/L = 256$。最终通过连接所有子层输出一个维度为 768（256 × 3）的新表示，这样，该层以较低的计算成本保存了大量的信息，也帮助权重向重要的 token 传递。

模型共需要 N 个深度连接层，以对应由注意力指导层产生的 N 个邻接矩阵，其中 N 为注意力头的个数。每个子层的图卷积计算用公式具体表示为：

$$\tilde{h}_{ti}^{(l)} = \rho\left(\sum_{j=1}^{n} \tilde{A}_{ij}^{(t)} W_t^{(l)} g_j^{(l)} + b_t^{(l)}\right) \tag{4.9}$$

其中，t 表示第 t 个头，$W_t^{(l)}$和$b_t^{(l)}$分别是可以学习的权重和偏差。

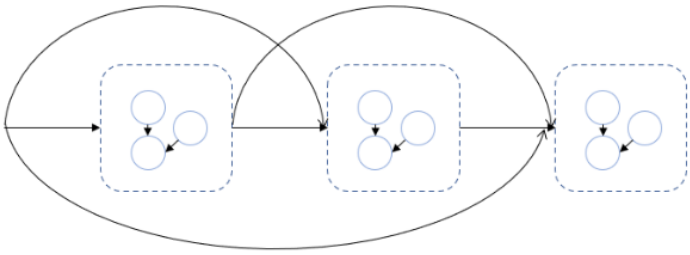

图 4.5 深度连接层网络结构

（4）线性组合层

该层将 N 个头对应的 N 个深度连接层的输出表示进行组合，得到最终输出:

$$\tilde{h}_{out} = W_{out}\tilde{h}_{in} + b_{out} \tag{4.10}$$

$$\tilde{h}_{in} = [\tilde{h}^{(1)}, \ldots, \tilde{h}^{(N)}] \tag{4.11}$$

$\tilde{h}_{out}$是 N 个注意力头组合而成的表示，W_{out} 和 b_{out} 是在训练期间需要学习的权重和偏差。

4.4.4 转喻分类器

这一层将最终的隐藏状态序列 H 映射到分类的类别上。表示 H_i 由第 i 个 token t_i输出， H_0表示 token 序列头部的“[CLS]”，作为整个句子序列的表示。

假设$\tilde{h}_{x'}, \ldots, \tilde{h}_{y'}$是针对句法融合层输出的实体 E 的词表示，x'和y'分别表示实体单词的起始索引和结束索引，并使用平均池化获得最终的实体编码:

$$H_e = \frac{1}{y' - x' + 1} \sum_{t=x'}^{y'} \widetilde{h_t} \tag{4.12}$$

为了实现转喻分类，将 H_0和 H_e串联起来，并在串联过后的表示上施加两个连续的全连接激活层，并应用 softmax 层进行最终预测：

$$H_{final} = \rho(W^*[\rho(W'\text{concat}[H_0; H_e] + b'] + b^*]) \tag{4.13}$$

$$\hat{\gamma} = argmax \frac{\exp(H_{final})}{\exp \sum_0^{|\Gamma|}(H_{final})} \tag{4.14}$$

$\hat{\gamma}$指转喻数据集中的其中一个类别，$|\Gamma|$是所有分类类型的个数，d 是隐藏向量表示的维度。

4.5 实验验证

4.5.1 实验数据集

为了评估模型的性能，本文在多个公开的英文转喻数据集和自建的中文转喻数据集上验证，除了上一章已经介绍过的 SemEval 英文基准转喻数据集，ReLocaR 英文转喻数据集，自建中文转喻数据集之外，我们额外增加了 WiMCor 数据集【28】进行大规模文本数据上的验证。

WiMCor 转喻数据集：大多数现有的转喻数据集在规模上有限，因此，有必要在大规模的转喻数据集上对模型进行评价。WiMCor 的训练集包含 92,563 个非转喻实例和 31,037 个转喻实例，其实例的平均长度是每句 80 个 token。由于硬件的限制，我们只使用训练数据集中的前 60,000 个样本实例和测试数据集中的前 10,000 个样本实例开展研究。

4.5.2 实验设置

（1）数据预处理

在依赖关系的获取上，模型使用 Stanford CoreNLP 【63】工具进行依存分析，并将所有依存树转换为对称邻接矩阵，为了简单起见，在转换中忽略了依赖的方向和类型。在前期的调研中，曾使用非对称矩阵进行了实验，但没有观察到实验效果上的改善。

在对于 BERT 编码模块的应用上，模型遵循 Devlin 等人【4】的方法，使用 BERT 中的分词器将单词分割成单词块（WordPiece）。为了适应每个数据集对应的序列长度分布，设置 SemEval 的最大序列长度为 256，ReLocaR 和 WiMCor 的最大序列长度为 128。

（2）对比实验设置

我们将本章提出的模型 EBERT+AGCN 与不同类型的模型比较，如基于特征工程的模型支持向量机 SVM，深度学习模型 BiLSTM 和预训练语言模型 BERT、ELMo 等。为了验证 EBERT+AGCN 权重模块的效果，在已有方法【17】的基础上，构建了模型 EBERT+GCN，并为它设置了合适的超参数。

（3）训练参数设置

对于所有基于 BERT 的模型，初始化 BERT 编码器的参数为 huggingface 发布的官方参数。通过在庞大的预训练语料库上实施两个无监督任务 Masked Language Model 和 Next Sentence Prediction 【4】，句子的 token 得到了很好的表征。实验结果表明，采用大小写敏感（cased）、包含 24 个 transformer 编码块的 BERT-LARGE 模型作为基线模型，对转喻数据集效果最佳。

对于 SemEval、RelocaR 和自建中文转喻数据集，设置批处理大小（batch size）为 8，训练 epoch 的数量为 20。对于 WiMCor 数据集，仅需训练 1 个 epoch 就达到了收敛。我们从集合{1,2,4,6,8}中选择多头注意的头数 N，从集合{5 × 10−6，1 × 10−5，2 × 10−5}中选择 AdamW lr 的初始学习速率。最终，(N = 8，lr = 1 × 10−5)、(N = 4， lr = 2 × 10−5)和(N = 4， lr = 2 × 10−5)，(N = 4， lr = 2 × 10−5)四种配置分别在 SemEval、ReLocaR、自建中文和 WiMCor 数据集上的效果达到最佳。

EBERT+AGCN 的训练使用了 Tesla v100-16GB GPU，内存消耗大约为 BERT-LARGE 的 1.5 倍。

4.5.3 对比模型

本节介绍本章实验中的对比模型，已在上一章中介绍过的模型不再介绍。

SVM+Wikipedia 模型：SVM+Wikipedia 是特征工程时代的 SOTA 模型，它将支持向量机应用于现有的维基百科网络上，使模型自动发现新的实体和关系。

LSTM 和 BiLSTM 模型：LSTM 是目前最流行的分类器之一【65】，由于具有记忆最后隐藏状态特征的功能，该模型获得了不错的结果，被广泛应用于各种 NLP 任务【66】。BiLSTM 在 LSTM 的基础上从两个方向感知来改进 token 表示**错误!未找到引用源。**，真正实现了应用上下文进行推理。此外，我们还分别对 GloVe 和 ELMo 两种不同的表示进行了测试，以确保模型结果的可靠性。

BERT-BASE/LG, +AUG, +MASK 模型： BERT 编码器可以将标记序列转换为深度分布式表示，Li 等人**错误!未找到引用源。**提出了部署 BERT-BASE 和 BERT-LARGE 的三种不同方式：

- BERT-BASE/LG: 仅使用一个不经过遮蔽的数据集进行微调。
- BERT-BASE/LG+AUG: 通过数据增强进行微调，将抽取的目标词随机替换为其他实体生成新的样本。
- BERT-BASE/LG+MASK: 在训练和测试的过程中，使用目标词遮蔽方法对输入的目标词进行了优化。在具体实现中，用单个标识符“[ENT]”替换了输入的目标词。

EBERT+GCN 模型：该模型采用硬句法策略，对所有依赖关系都采用相同的权重，通过图卷积网络将句法信息整合到词向量表示中。

EBERT+AGCN 模型：EBERT+AGCN 是在在 EBERT+GCN 的基础上构建的基于注意力指导的图卷积网络句法约束模型。EBERT+AGCN 中的注意引导层采用一种软注意力机制，为所有依赖关系分配适当的权重，来去除关系中的噪声。

图 4.6 用以说明本文中使用的所有 BERT 变体的具体实现方案，包括 BERT、

EBERT、EBERT+GCN 和 EBERT+AGCN。

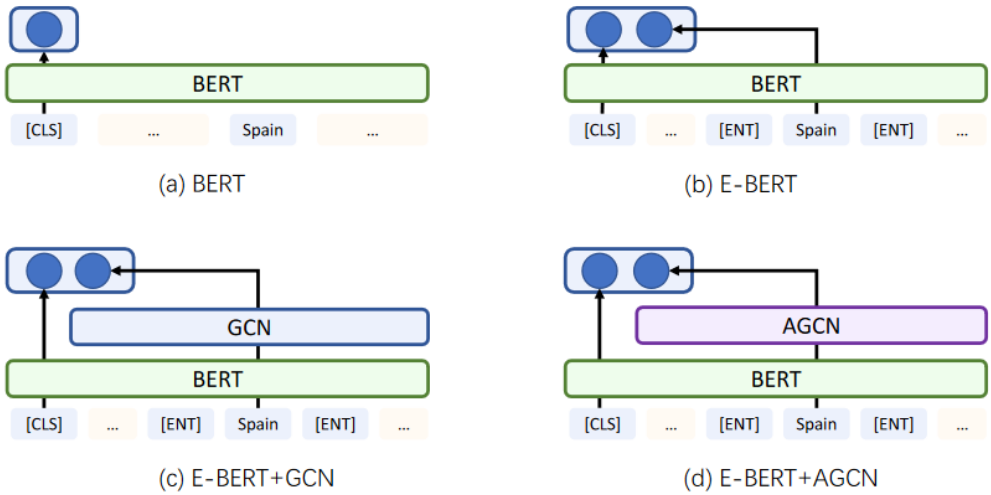

图 4.6 基于 BERT 的模型间结构对比

（a）BERT 模型使用了标准的转喻文本 token 序列作为输入，并依赖于“[CLS]”对应的向量表示进行转喻识别。

（b）EBERT 模型在 BERT 的基础上增加了实体指示器“[ENT]”和转喻实体对应的实体向量表示。

（c）EBERT+GCN 模型在 EBERT 基础上增加了 GCN 模块，以硬句法约束的方式融入句法知识。

（d）EBERT+AGCN 模型在 EBERT 基础上增加了 AGCN 模块，以软句法的方式融入了句法知识，拥有了为句法依存关系分配权重的能力。

4.5.4 实验结果与分析

（1）自建中文转喻数据集的句法模型实验结果

表 4.1 自建中文转喻数据集的句法模型实验结果

模型	Acc	Precision	Recall	F1-L	F1-M
EBERT	85.3	85.1	85.0	78.4	**91.9**
EBERT+GCN	86.5	86.6	86.6	**90.8**	82.6
EBERT+AGCN	**87.4**	**87.4**	**87.4**	88.3	86.5

表 4.1 中展示了 EBERT+AGCN、EBERT 和 EBERT+GCN 在中文转喻数据集上的实验结果。相比于无句法约束的 EBERT 模型和使用硬句法约束的

EBERT+GCN 模型，EBERT+AGCN 在各项指标上都得到了比较大的提升，验证了我们的模型在中文数据集上的有效性。但是由于中文本身的语言特性，上下文对于转喻词的影响看起来要比英语低一些。

（2）英文数据集的句法模型实验结果

为了进一步验证本文提出的基于注意力指导的图卷积网络句法约束模型的效果，在 SemEval、ReLocaR 和 WiMCor 英文数据集上进行了对比实验。

在 SemEval 和 ReLocaR 数据集上的结果如表 4.2 所示，表中展示了转喻识别模型的非转喻类 F1 值、转喻类 F1 值和准确率，“+NER+POS”指在基线模型上集成 NER 和 POS 特征。

首先，在准确率（Acc）比较中，基于特征工程的 SVM+Wikipedia 模型的性能仍然优于目前大多数深度神经网络模型，但大量手工特征所花费的人力成本依然是该类模型的痛点问题；LSTM 系列模型通过变化建模方式、改进 token 表示和集成外部特征等方法，获得了更好的结果，值得注意的是，由于 ELMo 表示已经提供了足够的语义，命名实体识别（NER）和词性识别（POS）特征提供的语义是冗余的，导致这两种特征对 BiLSTM （ELMo）模型的提升较小；PreWin 句法模型在 RelocaR 和 SemEval 数据集上的表现大大超过了基线模型 LSTM (GloVe)，在某种程度上证明了句法依存融入的重要性。

与以往的工作相比，EBERT+AGCN 具有较大优势，且取得了当前最好的结果。具体分析可得，EBERT+AGCN 明显优于基于特征工程的 SVM+Wikipedia 模型，同时还超越了包括 LSTM、BiLSTM 和 PreWin 在内的深度学习模型。此外，我们将 EBERT+AGCN 与两种我们构建的对比模型进行了比较：EBERT（无句法约束的实体感知 BERT 模型）和 EBERT+GCN（硬句法约束的实体感知 BERT 模型）。在 SemEval 和 ReLocaR 数据集上，EBERT+GCN 的准确率分别比 EBERT 提高了 0.3%和 0.2%，说明图卷积网络可以从句法依赖中捕获被忽略的信息，从而在一定程度上提高模型性能。使用 Immediate 5 方法对 EBERT+GCN 进行剪枝对实验效果影响不大，随机选择区域，无目的地对依赖关系剪枝可能会产生反效果。EBERT+AGCN 在 SemEval 和 ReLocaR 数据集上比 EBERT+GCN 分别提高了 0.7%和 0.2%，说明基于多头注意力的权重机制的引

入有助于图卷积网络更好地聚合信息。另外，EBERT+AGCN 的标准差（std）也更低，意味着模型的测试和运行更为稳定。

表 4.2 英文数据集的句法模型实验结果

模型	SemEval			ReLocaR		
	F1-L	F1-M	Acc(std)	F1-L	F1-M	Acc(std)
SVM+Wikipedia	91.6	59.1	86.2(N/A)	-	-	-
LSTM(GloVe)	85.2	28.7	72.6(1.48)	78.4	78.4	78.4(0.91)
+NER+POS	87.5	27.3	77.4(1.34)	80.6	80.6	80.6(0.92)
BiLSTM(GloVe)	83.2	37.4	75.4(1.72)	82.9	83.0	82.9(0.85)
+NER+POS	88.8	37.7	82.0(1.36)	84.2	84.2	84.2(0.69)
BiLSTM(ELMo)	91.9	54.7	86.3(0.45)	90.0	90.1	90.0(0.40)
+NER+POS	91.6	55.6	86.1(0.47)	90.1	90.1	90.1(0.36)
Paragraph	-	-	81.3(0.88)	-	-	80.0(2.25)
Immediate 5	-	-	81.3(1.11)	-	-	81.4(1.34)
Immediate 10	-	-	81.9(0.89)	-	-	81.3(1.44)
PreWin(GloVe)	90.6	57.3	83.1(0.64)	84.4	84.8	83.6(0.71)
PreWin(BERT)	-	-	87.1(0.54)	-	-	92.2(0.48)
BERT-BASE	-	-	85.0(0.46)	-	-	81.5(0.54)
BERT-BASE +AUG	-	-	84.5(0.85)	-	-	91.0(0.72)
BERT-BASE +MASK	-	-	87.1(0.89)	-	-	93.9(0.52)
BERT-LG	-	-	84.7(0.71)	-	-	91.3(0.54)
BERT-LG +AUG	-	-	85.0(1.10)	-	-	91.4(0.86)
BERT-LG +MASK	-	-	88.2(0.61)	-	-	94.4(0.31)
Ensembled BERT-LG +MASK	-	-	89.1(N/A)	-	-	94.8(N/A)
BERT-LG +MARK	93.5	60.0	87.6(0.55)	94.0	94.0	94.0(0.58)
BERT-LG +MARK+ENT	93.2	66.0	88.8(0.63)	95.2	95.3	95.3(0.44)
EBERT+GCN	93.5	67.5	89.1(0.60)	95.5	95.5	95.5(0.46)
EBERT+GCN Immediate 5	93.6	65.7	89.0(0.50)	95.3	95.4	95.4(0.44)
EBERT+AGCN	**94.0**	**68.3**	**89.8**(0.85)	**95.7**	**95.7**	**95.7**(0.34)

表 4.2 中给出了非转喻和转喻类的 F1 值，由于 ReLocaR 的非转喻和转喻分别占训练数据集中所有样本的 50%，是一个类平衡的数据集，所以 ReLocaR 的 F1 值比 SemEval 更高。ReLocaR 数据集的基线模型的 F1 值已经相对处于较高水平，EBERT+AGCN 较 EBERT 和 EBERT+GCN 只是略有提升。相反，SemEval 数据集的非转喻和转喻分别占 80%和 20%，这种不平衡导致转喻类语料的缺乏，使模型不能充分地学习语义特征。早期模型，如 LSTM，在转喻类上的 F1 值明显低于非转喻类，而 BERT 和 EBERT+AGCN 在 F1-M 指标上存在显著的性能差距，这表明 EBERT+AGCN 模型挖掘了额外的句法信息来处理语料不足的限制。总而言之，F1 值的对比显示 EBERT+AGCN 在 SemEval 和 ReLocaR 中都达到了最好的效果，且能够同时适应语料充足和不足的情况。

表 4.3 WiMCor 数据集上的实验结果

模型	WiMCor					
	Acc	F1	F1-L	F1-M	Precision	Recall
EBERT	90.3	79.8	95.8	74.5	86.0	74.5
EBERT+GCN	91.3	81.7	97.0	74.8	89.9	75.0
EBERT+AGCN	**91.8**	**83.0**	**96.5**	**78.2**	**88.5**	**78.2**

此外，为了验证模型在海量数据上的表现，额外使用 WiMCor 数据集进行实验。如表 4.3 所示，EBERT+AGCN 在超大数据量的情况下依旧能保证模型的鲁棒性，同时可以发现，硬句法约束改善了转喻识别任务的效果，但软句法约束对转喻识别任务的效果提升更明显。

（3）不同句子长度分布下模型效果对比实验

该实验比较了 EBERT+AGCN 和 EBERT 在不同句子长度下的准确性，如图 4.7 所示。实验在 SemEval 和 ReLocaR 数据集上进行，由于 ReLocaR 的平均句子长度比 SemEval 短，为了得到更为清晰的结论，主要关注在 SemEval 数据集上的模型表现。

在实验中我们发现转喻文本句子过长很可能影响分类的准确性，原因如下：

- 长句的上下文语义更难以捕捉和表示。
- 某些决定实体词是否转喻的关键词，如谓词（predicate），附近信息会更为嘈杂，不容易被表示。

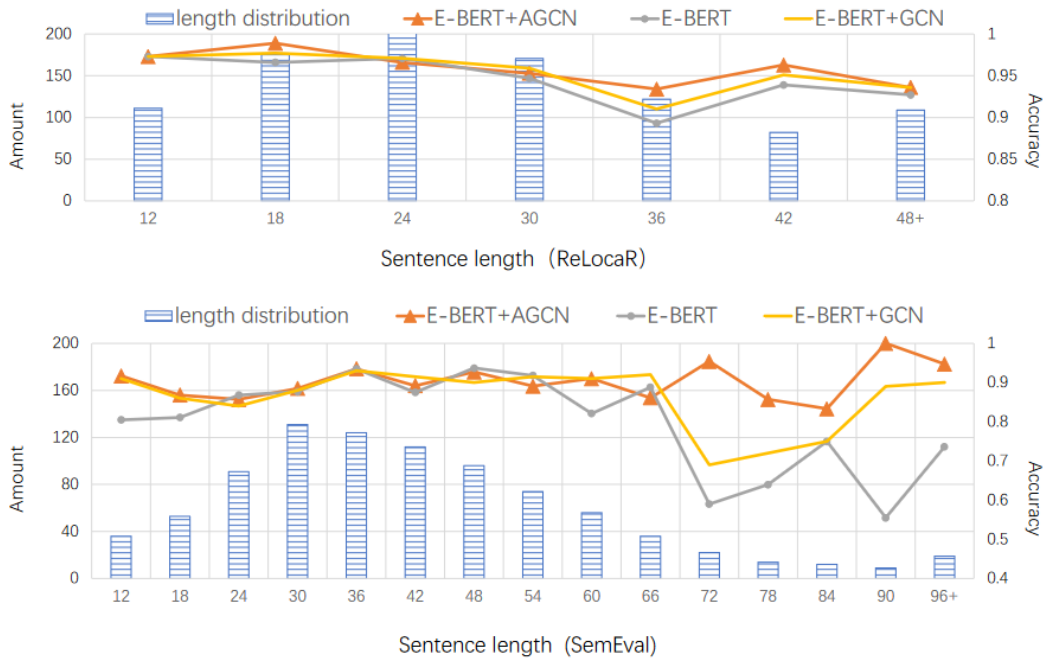

图 4.7 不同句子长度分布下模型效果对比实验

因此，基于序列的模型很难保证在长难句上的性能，一些研究表明，BERT 模型缺乏全局句法关系的可解释性，例如，Tang 等人【68】论述了 BERT 在建模长距离句法关系方面的不足。分析图 4.7 可得，随着句子长度的增加，BERT 的准确率有所下降，基于依存句法的模型可以在一定程度上解决此问题，并降低其计算复杂度，但是，虽然基于依存句法树的结构信息可以提高模型在很多 NLP 任务上的效果，BERT 却不能有效利用此类特征，而 EBERT+AGCN 却依靠强大的句法依存表示能力缓解了性能的下降。我们观察到，EBERT+AGCN 在所有句长分布上都优于 EBERT，并且随着句子长度的增加，效果的提升变得更加显著。综合实验结果，图 4.7 证实了基于注意力指导的图卷积网络句法约束模块 AGCN 帮助模型克服了在长难句上转喻识别率低的缺陷。

（4）不同训练 epoch 下模型效果对比实验

为了研究句法约束模块如何影响模型整体性能，该实验比较了 EBERT、EBERT+GCN 和 EBERT+AGCN 在 SemEval 和 RelocaR 数据集上迭代了 20 个 epoch 的 F1 值和准确率值，如图 4.8 所示。

通过 ReLocaR 数据集上的比较可得，在前几个 epoch 中，体量较轻的 EBERT 和 EBERT+GCN 模型获得了更好的 F1 和准确率，另一方面，由于 EBERT+AGCN 的神经网络结构较为复杂，模型参数较多，导致早期的 F1 值和准确率较低。

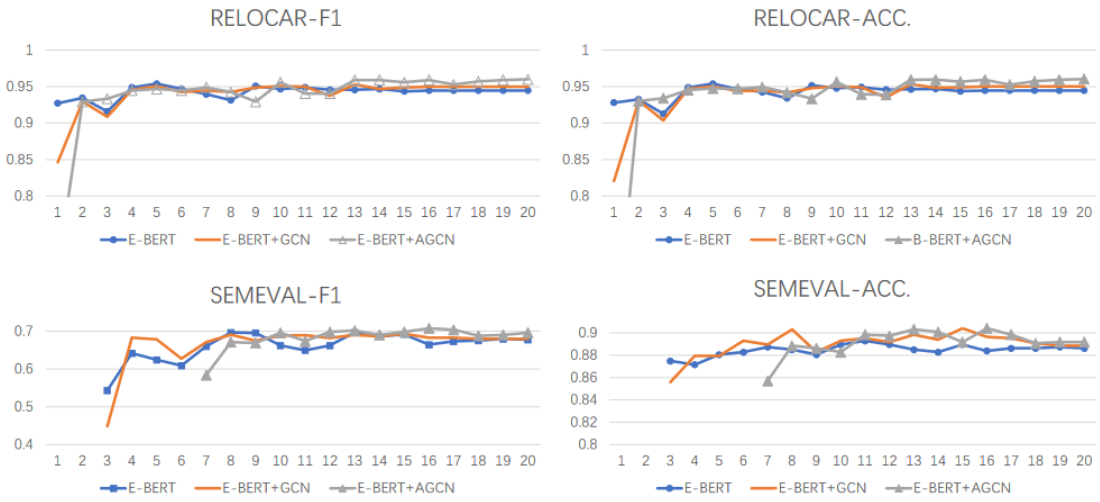

图 4.8 不同训练 epoch 下模型效果对比实验

经过数个 epoch 之后，各个模型都趋于稳态，模型之间的性能差别仍不明显。而当 epoch>15 后，EBERT+AGCN 的性能始终优于 EBERT+GCN 和 EBERT，证实了经过权重分配的句法信息在逼近收敛时显著提高了模型的表现。

观察 SemEval 数据集也可得和 ReLocaR 类似的结果，所以，在不同数据集上模块带来增益是较为稳定的。由于长难句多和数据不平衡的特性，SemEval 相比于 ReLocaR 需要更多的 epoch 来实现性能的稳定。

（5）不同数据量下模型效果对比实验

图 4.9 显示了 EBERT、EBERT+GCN 和 EBERT+AGCN 在不同数据量下的模型效果对比。由于 SemEval 数据集中的转喻样本较少，在数据量减少时准确率下降较多，研究意义不大，所以只在 ReLocaR 上进行实验。实验设置训练数据的数量为原始数据集的{20%，40%，60%，80%，100%}，研究后发现，即使只使用 20%的训练数据，三种模型也可以获得接近 90%的 F1 值，因此，EBERT+GCN/AGCN 模型对训练数据集的大小并不敏感。

图 4.9 不同数据量下模型效果对比实验

从模型的角度入手，在相同规模的训练数据集下比较，EBERT+AGCN 始终优于 EBERT，在所有的数据量设置下，EBERT 和 EBERT+AGCN 的性能差异始终大于 0.4%，当数据量更小时，EBERT+AGCN 的优势更明显，泛化能力更突出。

（6）网络深度对比实验

该实验将额外的 M 个句法模块堆叠在句法整合层已有的 2 个非注意模块和 2 个注意模块后，如表 4.4 所示。注意力模块的数量增加后模型性能显著下降，盲目建立更复杂的注意力模型起到了消极作用。如表 4.5 所示，在深度连接层中调整了 L_p（非注意模块中的子层数）和 L_n（注意模块中的子层数）后，模型准确率下降，说明了减少子层数量会导致模型在句法整合过程中丢失信息。

表 4.4 增加 M 个句法模块后模型结果对比

EBERT+AGCN	ReLocaR（Acc）	SemEval（Acc）
M=0（初始设置）	**95.7**	**89.8**
M=1	95.2	88.9
M=2	95.2	88.3

表 4.5 不同子层数下模型结果对比

EBERT+AGCN	ReLocaR（Acc）	SemEval（Acc）
L_p=2，L_n=4（初始设置）	**95.7**	**89.8**
L_p=1，L_n=2	95.6	89.2
L_p=1，L_n=1	95.3	89.3

（7）不同注意力头数量下模型结果对比实验

在 ReLocaR 数据集上，不同注意力头数之间的差距并不明显，但 $N=4$ 下的模型的结果优于其他所有 N 取值，F1 值达到了 95.8%。SemEval 数据集上，增加注意力头数可以显著提高模型性能，当 $N=8$ 时，得到最佳的 F1 值和准确率。表 4.6 展示了多头注意力模块中设置注意力头的数量 N 对模型结果的影响。

表 4.6 不同注意力头数量下模型结果对比

EBERT+AGCN 模型	ReLocaR		SemEval	
	Acc	F1	Acc	F1
N=1	95.6	95.5	89.4	70.1
N=2	95.6	95.6	89.4	68.3
N=4	**95.8**	**95.8**	89.8	70.5
N=6	95.6	95.6	89.3	70.3
N=8	95.7	95.7	**89.8**	**70.8**

综上，根据我们的设想，更多的注意力头本应可以捕捉不同单词的各种特征来进一步提高性能，但从实际实验的结果来看，使用更多的注意力头不能带来更好的效果，随着头数的增加，神经网络倾向于过拟合，使得模型性能趋于稳定或下降。

（8）实例研究

该实验通过一个实例的研究，从模型内部剖析句法融入的作用。给定转喻文本 S“他后来去指导**马来西亚**一年（He later went to manage **Malaysia** for one year）”，容易将“马来西亚（Malaysia）”与“马来西亚的分公司”或“马来西亚的球队”的概念联系起来，从而将“马来西亚”判断为一个转喻词。而由于动词短语“去（went to）”的指示作用极强，无句法约束模型忽略了文本中真正的谓词“指导（manage）”，即由于日常生活中“去某个地方”的习惯用法，无

句法约束模型会错误地将“马来西亚”视为非转喻的地点。以下通过模型的注意力权重的可视化解释上述问题如何在 EBERT+AGCN 模型中得到解决。

如图 4.10 所示为转喻文本 *S* 的注意力矩阵可视化图，其中：

- (a) EBERT 模型中 Transformer 编码器的平均注意力权重。
- (b) EBERT+AGCN 模型中的 Transformer 编码器的平均注意力权重。
- (c) EBERT+GCN 和 EBERT+AGCN 中的非注意模块的平均注意力权重。
- (d) EBERT+AGCN 中的注意模块的平均注意力权重。

该实验中，(a)和(b)对比了句法约束信息融入对 BERT 编码器的影响，(c)和(d) 比较了软句法约束和硬句法约束之间的效果差异。

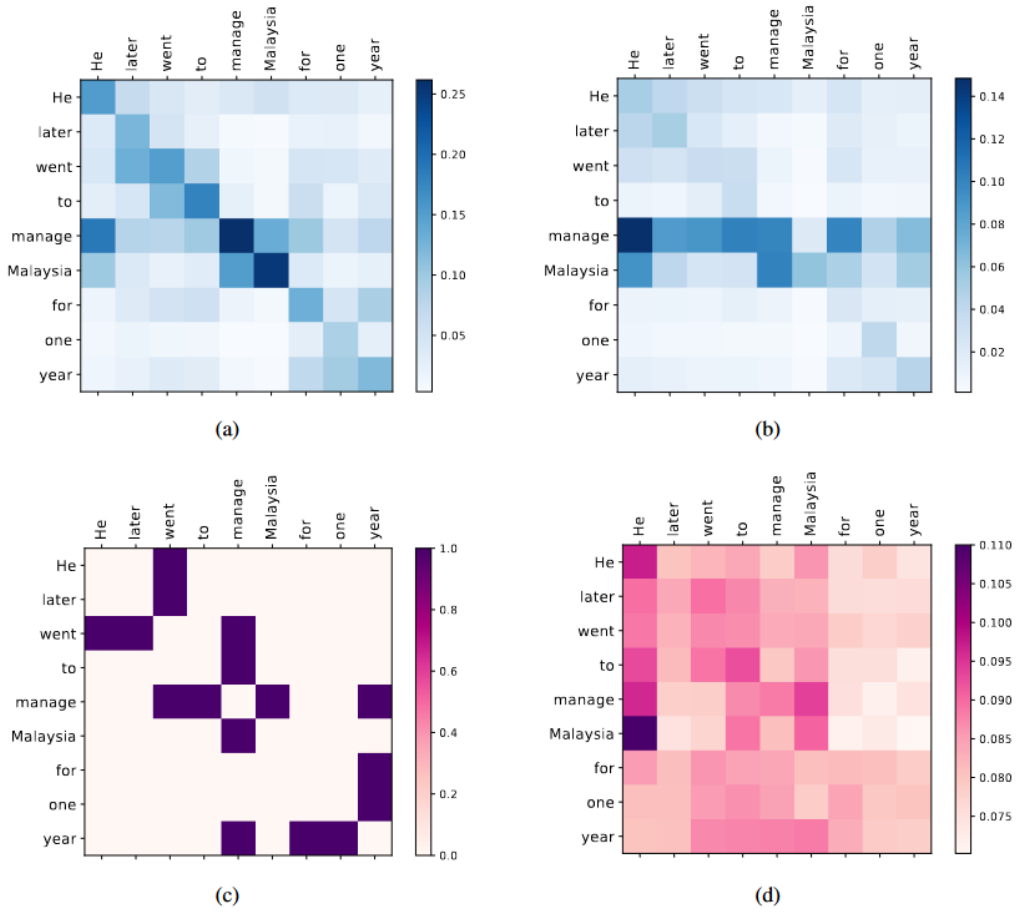

图 4.10 模型注意力矩阵可视化

首先比较 EBERT 中 Transformer 编码块的注意力矩阵，以检测句法集成对整个模型的贡献。分析(a)可得， EBERT 提供的语义知识不够，EBERT 无法分辨应该被着重考虑的目标词，导致 EBERT 中 token 的权重是分散的。在(b)中，EBERT+AGCN 的注意力更集中在谓词“指导”和实体“马来西亚”上，而不是

“去”或其他不相关的token，添加句法模块AGCN后，融入依存关系信息帮助模型理清文本的句法结构，使EBERT+AGCN更好地选择必要的token并丢弃不相关的或者误导的信息。

为了说明软硬句法约束之间的差异，先对文本 *S* 的句法结构进行如下分析。文本 *S* 可以分为主句“他后来去指导马来西亚（He later went to manage Malaysia）”和介词从句“一年（for one year）”。很明显，主句中包括谓词和实体，在转喻识别任务中占主导地位，提供大量有效信息，而硬句法约束方法认为“一（one）”与“年（year）”之间的修饰关系与涉及谓词或实体的关系具有相同的权值，这个过程显著引入句法噪声。

注意力指导层将硬依存邻接矩阵转换为软注意力矩阵，允许句法模块选择相关的句法特征。此外，与 Transformer 编码层中的注意力矩阵不同，注意力指导层中的注意力权重反映了依赖关系的重要性。因此，我们将注意力指导层的矩阵可视化，以证明软注意力矩阵的优越性。如图 4.10 的(d)所示，EBERT+AGCN 中，从句“一年（for one year）”虽然存在依存关系，如“一”与“年”之间的修饰关系，但在权重分配的作用下，此类依存关系的权重与主句相比明显较低，而多头注意力机制的加入，使得该模型无需预先设定剪枝策略，自动且准确地获得高质量的关系。

4.5.5 错误分析

本节使用几个典型的样例从多个方面分析EBERT+AGCN模型与EBERT模型的优势和劣势，如表4.7所示。

表中粗体表示例句的实体。“分类”为目标实体的真实类别标签，“√”和“×”表示EBERT+AGCN和EBERT判断的正误。

（1）语义孤立

EBERT+AGCN通过依存关系预测实体类型，在例句S1中，短语“2015年马赛”与其上下文没有明确的语义关系，其语义与上下文产生了分割，在语义孤立的情况下，造成了“马赛”与上下文之间的依存关系缺失，使EBERT+AGCN未能将“马赛”转喻为“马赛的一项体育赛事”。

表 4.7 句法模型错误分析

#	例句	分类	EBERT	EBERT+AGCN
S1	她的个人最好成绩是室外 1.92 米（2015 年马赛）和室内 1.93 米（2015 年布达佩斯）。	转喻	×	×
S2	意大利、法国、德国、奥地利、匈牙利、罗马尼亚和南斯拉夫执行了对敌人的军事、工业和运输的远程战略轰炸任务。	转喻	×	√
S3	这导致了石勒苏益格-荷尔斯泰因的德国人公开起义，他们支持从丹麦独立出来，并与德意志联邦保持密切联系。	非转喻	×	×
S4	这张唱片到 1965 年 9 月又卖出了 50 万张，使其成为继 1958 年电影南太平洋的原声带之后，第二张在英国卖出 100 万的专辑。	转喻	×	√
S5	哈金森乘坐希尔斯伯勒前往澳大利亚，该船也被称为“发烧船”，因为船上 300 名囚犯中约有 95 人死于伤寒。	转喻	×	×

（2）谓词确定

在转喻识别任务中，谓词起着关键作用**错误!未找到引用源。**，如果一个实体 A 对其他实体施加动作，A 很可能是一个经过转喻的实体。S2 是一个典型的例子，其谓词与实体之间相距较远，导致没有句法约束的 EBERT 难以捕捉两者之间的句法或语义关系，EBERT+AGCN 联系了实体“意大利”和谓词“执行”之间的依存关系，从而正确预测了转喻类别。

在更复杂的情况下，EBERT+AGCN 也可能无法识别转喻。例如，在 S3 中，传统模型很容易找到谓词“起义”，而该动作的实施者是“石勒苏益格-荷尔斯泰因的德国人”，而非 S3 中的实体“石勒苏益格-荷尔斯泰因”。EBERT+AGCN 利用句法结构知识高效地跟踪谓词，却忽略了动作的实施者，一旦句法的复杂度超过了 EBERT+AGCN 的理解范围，依然会发生错误。

（3）知识缺失

在转喻文本中，许多转喻实体是表示著名作品或事件的专有名词。这些作品或事件的知识，通常被存储于外部知识库中，在某些 NLP 任务中得到应用，而转喻识别模型还没有引入先验知识的先例，导致可解释性较差。比如 S4 中，“南太平洋”指的是 1958 年的一部电影，EBERT 未能正确识别，而得益于“电影”与“南太平洋”之间的依赖关系，EBERT+AGCN 成功地将“南太平洋”的含义扩展到了电影《南太平洋》。S5 则是一个 EBERT+AGCN 识别失败的例子，尽管文中提到了“希尔斯伯勒”指代的是“发烧船”，但是由于文本依存关系复杂，EBERT+AGCN 没有检测到转喻，如果有外部先验知识指明“南太平洋”和“发烧船”之间的潜在联系，将大大提高转喻效率。

4.6 本章小结

本章对转喻识别的句法融入问题进行研究。针对转喻文本长句识别难，谓词识别不清以及缺乏句法依赖权重分配方法的问题，提出了一种基于注意力指导的图卷积网络句法约束模型，通过图神经网络，完整且准确地融入句法依赖知识；同时，对不同的依赖关系进行权重的调配，不仅获得了抵御噪声干扰的能力，而且进一步提升了转喻识别的准确率。最终，所提出的方法和模型能够有效地解决转喻文本中长难句准确率下降问题和关键词识别问题，提高了模型的文本分析能力。

第五章 基于变分信息瓶颈的语义压缩技术模型

5.1 引言

正确识别现实场景中的转喻，前提是对转喻文本的真实语义有准确全面的理解。得益于深度学习的蓬勃发展，转喻模型在模型架构上越来越完善，却始终没有办法解决训练文本和现实语料脱节的问题。当用较少的训练数据进行微调时，模型倾向于通过训练数据的出现频率进行判断，最终导致过拟合问题。另外，多数转喻文本有信息冗余及上下文语言表述多样的特点，造成语义特征抽取的困难。所以，从模型的泛化性和鲁棒性的角度来考虑，抽取文本语义概念框架并对文本进行语义压缩尤为必要。

本章拟解决以下两个子问题：1）如何增强模型对未知文本数据的敏感度，解决过拟合的问题；2）如何构建一种语义压缩方法，解决语义信息冗余，实现文本的去噪和泛化。

5.2 相关工作

5.2.1 变分自编码器

变分自编码器（VAE）是一种有向图模型，也是深度学习领域中一个流行的深度生成框架【69】**错误!未找到引用源。**，提供了一种基于概率的方式观察潜在空间，提升模型的可扩展性和训练稳定性。假设观察到的变量 x 是一个来自底层过程的随机样本，其真实分布 $p_*(x)$是未知的，那么可以使用带有参数θ的参数化模型 $p_\theta(x)$来近似这个底层过程，VAE 的生成过程遵循$p(x|z)p(z)$形式的贝叶斯网络:

- 从先验分布 $p_\theta(z)$生成一组潜在变量 z;
- 给定生成分布 $p_\theta(x|z)$，在 z: $z \sim p_\theta(z)$，$x \sim p_\theta(x|z)$的条件下，重构输入 x。

在 VAE 中引入潜在变量 z 以高效利用先验知识，有助于识别未在训练过程中出现的数据。由于真实的后验概率 $p_\theta(z|x)$是未知的，Kingma 和 Welling【69】

巧妙地引入了后验概率分布q_ϕ(z|x)，也就是“recognition”模型，来近似真实的后验概率 p_θ(z|x)，这时，研究的问题转向了最大化证据下界（evidence lower bound，ELBO），而不是直接对难以处理的边际对数似然进行极大似然估计：

$$\log p_\theta(\mathrm{x}) = \underbrace{\mathbb{E}_{q_\phi(\mathrm{z}|\mathrm{x})}\left[\log p_\theta(\mathrm{x},\mathrm{z}) - \log q_\phi(\mathrm{z}|\mathrm{x})\right]}_{\mathcal{L}_{\theta,\phi}(\mathrm{x})(\text{i.e., ELBO})} + \underbrace{\mathbb{E}_{q_\phi(\mathrm{z}|\mathrm{x})}\left[\log q_\phi(\mathrm{z}|\mathrm{x}) - \log p_\theta(\mathrm{z}|\mathrm{x})\right]}_{D_{KL}(q_\phi(\mathrm{z}|\mathrm{x})||p_\theta(\mathrm{z}|\mathrm{x}))} \tag{5.1}$$

也可以写成：

$$\mathcal{L}_{\theta,\phi}(\mathbf{x}) = E_{q_{\phi(\boldsymbol{z}|\boldsymbol{x})}}[\log p_\theta(\boldsymbol{x}|\boldsymbol{z})] - D_{KL}(q_{\phi(\boldsymbol{z}|\boldsymbol{x})}||p_\theta(z)) \tag{5.2}$$

上式中右边的第一项为重构误差，第二项为 p_θ(z)与 p_θ(x|z)之间的 KL 散度（KLD），KLD 是一种使q_ϕ(z|x)接近 p_θ(z|x)的正则化过程。θ为生成模型参数，ϕ为推理模型参数。为了使误差通过潜在变量反向传播，Kingma 和 Welling【69】使用辅助噪声变量ϵ的可微变换$g_\phi(\epsilon,\mathrm{x})$重新为潜在变量 $\mathrm{z} \sim q_\phi(\mathrm{z}|\mathrm{x})$设置参数，使之符合单位高斯先验分布，即：

$$z \sim q_\phi(z|x) = g_\phi(\epsilon, x)\ and\ \epsilon \sim p(\epsilon) = \mathcal{N}(\theta, I) \tag{5.3}$$

VAE 的训练过程是无监督的，将重构输入 x 作为目标函数。大多数的下游任务都需要标签的监督，由于 z 定义的潜在空间可以捕获数据相关的有效信息，VAE 也可以解决标记数据上的监督学习问题，但是，由于 VAE 在训练过程中不涉及 y 标签，因此训练步骤和推理步骤是分离的，仅仅使用 VAE 编码器 x→z 和辅助解码器 z→y 进行预测。

VAE 用基于概率的方式描述潜在空间中的对象，增强了可扩展性和训练稳定性，然而，基于 VAE 的模型往往会产生相对模糊的表示，无法捕捉细节，因此β-VAE 更倾向于使用 InfoGAN **错误!未找到引用源。**，这时，可以存储的最大信息表示为：

$$\mathcal{L}_{\theta,\phi}(\mathbf{x}) = E_{q_{\phi(\boldsymbol{z}|\boldsymbol{x})}}[\log p_\theta(\boldsymbol{x}|\boldsymbol{z})] - \beta \times D_{KL}(q_{\phi(\boldsymbol{z}|\boldsymbol{x})}||p_\theta(z)) \tag{5.4}$$

其中β是拉格朗日乘数。该函数的目标是最大化潜在瓶颈 z 和标签 y 之间的潜在信息，同时丢弃所有与 y 无关的信息。（5.4）中的负 KL 散度项可以缩小两者之间的差距，所以可以考虑在目标函数的负 KL 项上分配更多的权值，以减少在$\beta \geq 0$条件下训练和测试时潜变量编码的差异。β-VAE 认为后验分布q_ϕ(z|x)是信息瓶颈，后验分布通过最小化 KL 项和最大化数据对数似然有效地传递输

入 x 的信息。因此，β-VAE 的目标 $D_{KL}(q_{phi}(z|x)||p(z))$的 KL 散度项可以作为每个数据样本通过潜在通道传输的信息量的上界，而当$q(z|x) = p(z)$时，KL 散度为零。

5.2.2 变分信息瓶颈

在大多数情况下，我们更多地关注于如何学习一个判别模型 $p_\theta(y|x)$，而不是一个自我重构任务 $p_\theta(x)$，其中，x 是一个输入句子，y 是我们希望预测的类标签。假设 z 是潜在变量，当输入 x 进行通道变换，并得到一个中间输出 z，在已知数据处理不能增加信号$I(x;y) \leqslant I(x;z)$中包含的信息的情况下，通过某种操作获得类似 x→z→y 的信号。

给定数据处理不等式，可以对数据进行标识编码（z = x）来确保表示的信息最大化，然而，在这种数据表示不能被随意使用的情况下，需要找到在复杂度的约束下能得到的最优表示，例如，编码数据和原始数据之间的互信息$I(x, z) \leqslant I_c$就是一个简单有效的约束，其中I_c是信息约束，目标为:

$$\max_{\theta} I(\boldsymbol{z}, \boldsymbol{y}; \theta) \quad \text{s.t.} \quad I(\boldsymbol{x}, \boldsymbol{z}; \theta) \leq I_c \tag{5.5}$$

其中I是互信息，函数的目标是最大化潜在瓶颈 Z 和任务 Y 之间的潜在信息，同时抛弃所有可能出现在输入中与 Y 无关的信息。

变分信息瓶颈（VIB）【72】是一种基于互信息的信息理论监督表示框架。VIB 在最大化潜在变量 z 和标签 y 之间的互信息$I(z, y; \theta)$的同时最小化 z 与 x 之间的互信息。

$$R_{VIB}(\theta) = I(\boldsymbol{y}, \boldsymbol{z}; \theta) - \beta \times I(\boldsymbol{x}, \boldsymbol{z}; \theta) \tag{5.6}$$

由于互信息的难解性，采用变分方法来降低目标的下界，于是，VIB 的最终目标如下所示：

$$\tilde{\mathcal{L}}_{\theta,\phi}(\boldsymbol{y}, \boldsymbol{x}) = E_{q_\phi(\boldsymbol{z}|\boldsymbol{x})}\left[\log p_\theta(\boldsymbol{x}|\boldsymbol{z})\right] - \beta \times D_{KL}(q_{\phi(\boldsymbol{z}|\boldsymbol{x})}||p_\theta(z)) \leq R_{VIB}(\theta) \tag{5.7}$$

所以，提升潜在通道的容量需要分散后验均值或减少后验方差，而这两种方法都会增加 KL 散度。

VAE 和 VIB 非常相似，它们的不同点在于 VIB 用一个分类器代替了 VAE 中的解码器网络，只保留多标签分类等判别任务所需的信息。VAE 和 VIB 都是通用的目的表示学习方法，在某些特定任务中，它们的效果并不稳定。

5.3 问题定义和分析

表 5.1 以“马云”为实体的转喻文本

例句	类别
马云收购饿了么。	转喻
马云在央视演讲。	非转喻
…	…
马云作为翻译首次访问美国。	非转喻
马云表示国内互联网市场庞大。	非转喻

目前的预训练语言模型在很多语境推理任务中仍然达不到令人满意的效果，一方面，这些模型有大量的参数，在微调过程中可能会造成过度拟合，而且，在这种预训练语言模型中学习到的大量特征可能与转喻识别任务本身无关。现实中转喻和非转喻的出现频率比为 1:4 左右**错误!未找到引用源。**，即生活中的大部分转喻文本中的实体都是未经过转喻的。以表 5.1 举例，大多数网络上的转喻文本中，“马云”指的是“马云”本人，此时不存在转喻的现象，而只有在例句“马云收购饿了么”中，才会被转喻为“马云”的公司“阿里巴巴”。

偏倚的样本分布是过拟合现象发生的“温床”，在实体“马云”大概率为非转喻的情况下，模型很难对较少出现的转喻现象施加正确的判断，判断的标准会逐渐变成转喻现象出现的概率，而不是转喻文本的语义本身。所以，开发一个可以适应不同数据质量和数据特性的模型能显著提高学习的成功率。

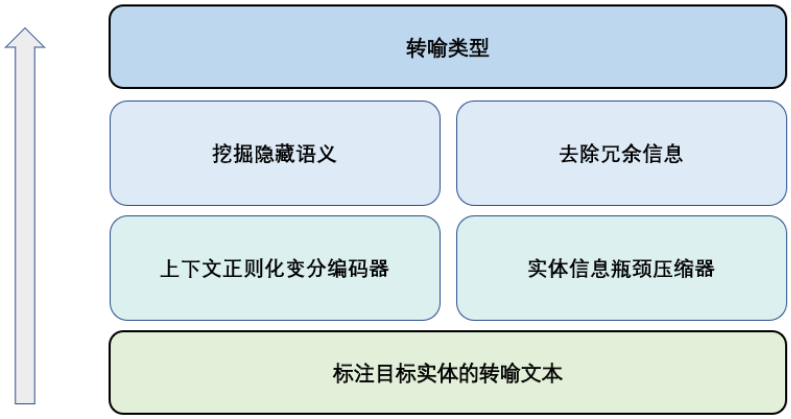

图 5.1 语义转喻识别方法整体设计

另一方面，由于标注数据非常耗时，标注的人力成本非常昂贵，所以大多数用于转喻识别的公开数据集的规模有限，且质量较差，文本中关键语义信息大量缺失，很难通过微调区分相关和不相关信息，需要开发一个转喻识别模型克服转喻文本的表达多样性和数据稀疏性。

因此，为了解决上述问题，本章就基于变分信息瓶颈的语义压缩技术进行研究，其识别方法整体设计如图 5.1 所示。首先从标注目标实体的句子中抽取出实体的表示，将上下文表示输入上下文正则化变分编码器中，实体表示输入实体信息瓶颈压缩器中，根据模型的输出结果预测出目标实体的转喻类型。

总结来说，我们工作的贡献在于：

- 使用正则化变分编码方法，挖掘隐藏的语义，防止过拟合。
- 使用瓶颈压缩方法，去除冗余信息，解决转喻文本表达多样化问题，实现语义知识的泛化。

5.4 模型架构

如图 5.2 为我们的模型 MIXINFBERT 的总体架构，模型由四个模块组成：EBERT 编码器（Encoder）、信息瓶颈压缩器（Compressor）、变分正则化器（Regularizer）、逻辑回归分类器（Classifier）。

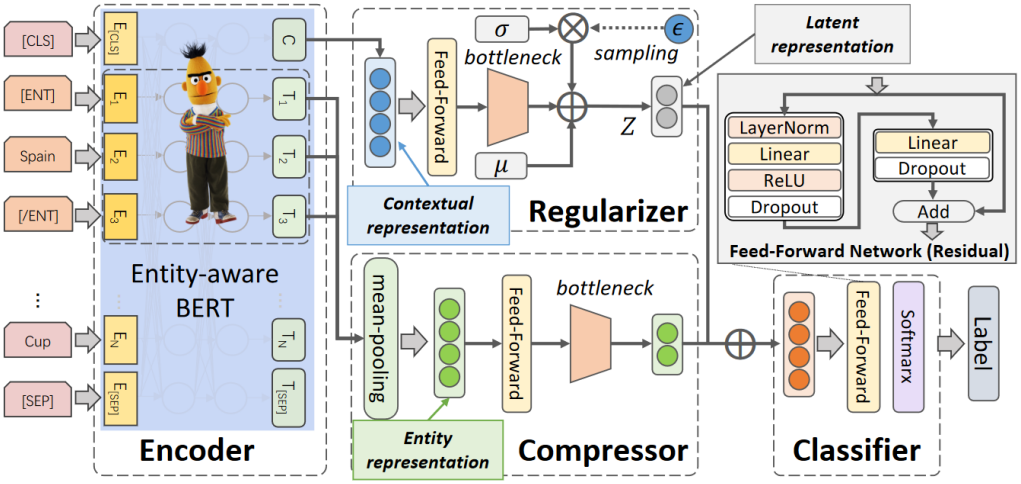

图 5.2 基于变分信息瓶颈的语义压缩技术模型 MIXINFBERT

模型从输入到输出的具体流程为：

① 将输入的句子输入到 EBERT 编码器来获得每个 token 的上下文表示；

② 将实体表示输入实体压缩器，以捕获有效的局部信息；

③ 将上下文表示输入上下文正则化变分编码器中，提取语义框架；

④ 将合并后的最终表示输入逻辑回归分类器中，识别转喻类别。

5.4.1 信息瓶颈压缩器（Compressor）

模块使用了一个具有朴素瓶颈神经结构的全连接前馈网络压缩器，该压缩器由两个带有 ReLU 激活函数的线性转换层组合而成：

$$\boldsymbol{e} = FFN(H_e) = \sigma(0, W_1 H_e + b_1)W_2 + b_2 \qquad (5.8)$$

其中，层与层之间使用不同的参数，虽然我们在模型中使用了全连接网络，但是也可以认为使用了两个内核大小为 1 的卷积，输入和输出的维度分别是 d_{input}=1024 和 d_{output}=256，而额外设置的内部层（inner-layer）维度 d_{inner}=1024。

5.4.2 变分正则化器（Regularizer）

MIXINFBERT 将后验分布$q(z|x)$作为上下文表示的信息瓶颈，其中，后验分布通过最小化β-weighted KL 项和最大化 log 似然有效传递数据点 x 的信息，而正则化器能够使后验分布与高斯先验相匹配。

研究表明，转喻识别不会建立在严格的规则之上，而发生在一个词汇违反其选择性限制或涉及语法上的错误用法时【8】。在转喻识别中，预测分类的隐含表示由两部分组成：相对固定的实体部分和相对可变的上下文部分。转喻的实体部分已经包含了足够的信息，无论实体是否转喻，这个信息非常稳定；相反，上下文部分非常嘈杂，含有大量无关特征。

为了去除原始高维表示中的冗余特征，我们以原始的混合信息瓶颈框架为基础，采用变分正则化方法，在训练目标中加入正则化。从信息论的角度出发，模型将q(z|x)看作是一组独立的加性高斯白噪声通道，每个通道都可以粗略地传输数据信息。在实际的测试过程中，假设 e 和 z 是相互独立的，保证 e 对后验项没有任何影响，从而可以被忽略。混合信息瓶颈框架的最终目标可以用下式表示：

$$\tilde{\mathcal{L}}_{\theta,\phi}(\boldsymbol{y},\boldsymbol{x};\boldsymbol{z},\boldsymbol{e}) = E_{q_{\phi}(\boldsymbol{z}|\boldsymbol{x})}[\log p_{\theta}(\boldsymbol{y}|\boldsymbol{z},\boldsymbol{e})] - \beta \times D_{KL}(q_{\phi}(\boldsymbol{z}|\boldsymbol{x})||p_{\theta}(z)) \tag{5.9}$$

z 为由上下文的高斯噪声通道得到的隐含表示，e 为从实体通道得到的压缩嵌入表示，表示H_i是由第 i 个 token t_i输出得到，H_0表示分词后的 token 序列头部的“[CLS]”对应的隐含向量，作为句子上下文向量的池化嵌入，并利用重参数化技巧得到上下文的变分隐含表示 z：

$$\mu_{\boldsymbol{h}}, \sigma_{\boldsymbol{h}} = \boldsymbol{M}(H_0), \boldsymbol{\Sigma}(H_0) \tag{5.10}$$

$$\boldsymbol{z} = \epsilon_{\boldsymbol{h}} \times \sigma_{\boldsymbol{h}} + \mu_{\boldsymbol{h}} \text{ and } \epsilon_{\boldsymbol{h}} \sim \mathcal{N}(0,1) \tag{5.11}$$

5.4.3 逻辑回归分类器（Classifier）

为了实现转喻分类预测，MIXINFBERT 首先将句子上下文向量H_0和实体向量 H_e 连接，并通过全连接层映射到所有分类类别总数的维度上，最后得到预测的结果：

$$H_{final} = \rho(W^* * [\rho(W'[\boldsymbol{e} \oplus \mathbf{z}] + b')] + b^*]) \tag{5.12}$$

$$\boldsymbol{y}^* = \arg\max \frac{\exp(H_{final})}{\exp \sum_0^{|\Gamma|}(H_{final})} \tag{5.13}$$

$\boldsymbol{y}^*$指转喻数据集中预设的其中一个类别，$|\Gamma|$是所有分类类别的个数，d 是隐藏向量表示的维度。

5.5 实验验证

5.5.1 实验结果与分析

本章的实验主要在英文转喻数据集 SemEval、ReLocaR、CoNLL 上进行验证。

对于所有的英文转喻数据集，设置句子的最大长度为 256，批处理大小（batch size）为 8，AdamW 的学习率为 2 × 10−5，训练 epoch 的数量为 10，dropout rate 为 0.2。

（1）英文数据集的语义模型实验结果

表 5.2 英文数据集的语义模型实验结果

模型	CoNLL			ReLocaR			SemEval		
	F1-L	F1-M	Acc(std)	F1-L	F1-M	Acc(std)	F1-L	F1-M	Acc(std)
SVM+Wikipedia	-	-	-	-	-	-	91.6	59.1	86.2(N/A)
Paragraph	-	-	-	-	-	80.0(2.25)	-	-	81.3(0.88)
Immediate-5	-	-	-	-	-	81.4(1.34)	-	-	81.3(1.11)
Immediate-10	-	-	-	-	-	81.3(1.44)	-	-	81.9(0.89)
PreWin(GloVe)	-	-	87.9(0.22)	84.2	84.8	83.6(0.71)	90.6	57.3	83.1(0.64)
PreWin(BERT)	-	-	92.6(0.32)	-	-	92.2(0.48)	-	-	87.1(0.54)
BERT	-	-	89.5(0.84)	-	-	91.3(0.57)	-	-	84.7(0.71)
BERT+AUG	-	-	-	-	-	91.4(0.86)	-	-	85.0(1.10)
BERT+MASK	-	-	93.9(0.54)	-	-	94.4(0.31)	-	-	88.2(0.61)
BERT+MARK	95.7	92.0	94.4(0.67)	94.3	94.5	94.5(0.58)	91.6	64.7	86.4(0.83)
EBERT	95.6	91.7	94.5(0.30)	95.0	95.1	95.1(0.39)	93.0	68.5	88.5(0.52)
MIXINFBERT	**95.8**	**92.2**	**94.7(0.26)**	**95.4**	**95.5**	**95.5(0.28)**	**93.1**	**70.6**	**89.2(0.44)**

表 5.2 展示了基于变分信息瓶颈的语义压缩模型在英文数据集上的实验结果，“F1-M”和“F1-L”为转喻类和非转喻类上的 F1 值，Acc 和 std 为模型的准确率和准确率对应的标准差。为了评估 MIXINFBERT 的转喻识别能力，对每个对比模型进行了 10 次以上的实验，并记录实验效果的平均值。

从表 5.2 可以看出，在三个数据集上，MIXINFBERT 都优于之前的 SOTA 模型 BERT+MASK 错误!未找到引用源。，分别在 CoNLL 上提高了 0.8%，ReLocaR 上提高了 1.1%，SemEval 上提高了 1.2%的准确率。

具体分析各个模型的准确率，可以发现，较为古老的特征模型 SVM+Wikipedia 的性能仍然优于近几年的基于 BiLSTM 的神经网络模型，如 Paragraph 模型、Immediate 模型和 PreWin（GloVe）模型，然而，该方法高度依赖于特征工程，花费大量时间和人力成本，阻碍了模型的进一步应用。PreWin（BERT）以一种惰性的方式生成固定的 token 嵌入并高效提取特征，在所有三个数据集上的表现都超过了 Paragraph、Immediate 和 PreWin（GloVe），间接证明了预训练语言模型对于转喻识别的重要性。

BERT 的表现优于 PreWin（GloVe）等传统深度学习模型，而 BERT+AUG 和 BERT+MASK 分别使用了数据增强和目标词屏蔽提高了 BERT 的性能。BERT+MARK 的结果表明，单纯地为实体添加标记也可以提高一部分模型的性能。

从表中关注到，MIXINFBERT 模型获得了最好的效果，超过了所有先前的模型，包括 Paragraph、Immediate、PreWin、BERT+MASK 和 EBERT。此外，MIXINFBERT 引入信息瓶颈机制更好地聚合了关键语义信息，效果和 EBERT 相比有了显著的提高。

从 F1 值来看，MIXINFBERT 在转喻类上的表现要比非转喻类差，但它在转喻类上仍然优于一切现有的模型，显示了它在挖掘额外语义信息层面的优势。

（2）模型收敛速度对比

本实验比较了三种模型在 ReLocaR 和 SemEval 数据集上的收敛速度，其中蓝色曲线，橙色曲线，绿色曲线分别代表 BERT+MARK，EBERT，MIXINFBERT 随着 epoch 提升的准确率变化图。如图 5.3(a)和图 5.3(b)所示，在刚开始训练的阶段，BERT+MARK 模型的准确率较低；在 epoch 0 到 epoch 2 阶段，MIXINFBERT 的准确率有比较大的提升；在 epoch 5 到 epoch 10 阶段，所有对比模型的准确率都相对稳定。分析实验结果，MIXINFBERT 虽然比基线模型有更多的训练参数，但没有减弱其训练的启动速度，在训练一开始就取得了较

好的效果，另一方面，在模型达到稳定后，其性能始终优于两个对比模型，体现了混合信息瓶颈方法轻量且高效的特性。

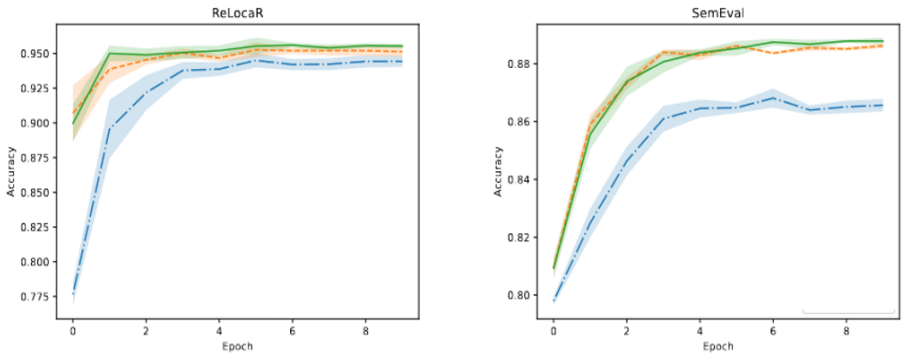

图 5.3 模型收敛速度对比

（3）语义模块消融实验

表 5.3 语义模块消融实验

模型	CoNLL			ReLocaR			SemEval		
	F1-L	F1-M	Acc(std)	F1-L	F1-M	Acc(std)	F1-L	F1-M	Acc(std)
MIXINFBERT	**95.8**	**92.2**	**94.7(0.26)**	**95.4**	**95.5**	**95.5(0.28)**	**93.1**	**70.6**	**89.2(0.44)**
去除 Compressor	95.7	92.0	94.4(1.12)	93.3	93.1	93.2(0.89)	89.1	52.4	82.3(0.91)
去除 Regularizer	95.6	91.7	94.2(0.25)	95.1	94.8	94.9(0.48)	93.0	70.2	88.6(0.28)

我们通过依次去除上下文正则化变分编码器（Regularizer）和实体信息瓶颈压缩器（Compressor）来验证两个模块对模型产生的影响，实验结果如表 5.3 所示。在不同的数据集上，各个模块的贡献是不同的，对于较大的数据集 CoNLL，Regularizer 对 CoNLL 的贡献比 Compressor 的贡献更显著，而对于较小的数据集 ReLocaR 和 SemEval，去除 Compressor 后，由于失去了对模型语义的去噪功能，模型性能大幅下降，相比于 Compressor，Regularizer 的正则化过程在较小数据集上只起辅助作用。总结来看，Regularizer 和 Compressor 都对模型的性能做出了一定的贡献。

（4）变分信息效果验证

该实验验证了模块在是否应用变分操作的情况下的性能差异。实验将变分信息瓶颈（VIB）替换为了普通信息瓶颈（IB），结果如表 5.4 所示，其中对比

模型共有四种：VIB+IB（上下文表示使用 VIB，实体表示使用 IB）、IB+VIB（上下文表示使用 IB，实体表示使用 VIB）、VIB+VIB（上下文表示使用 VIB，实体表示使用 VIB）、IB+IB（上下文表示使用 IB，实体表示使用 IB）。

表 5.4 变分信息效果验证

变分设置		CoNLL			ReLocaR			SemEval		
上下文	实体	F1-L	F1-M	Acc(std)	F1-L	F1-M	Acc(std)	F1-L	F1-M	Acc(std)
VIB	IB	95.8	**92.2**	**94.7(0.26)**	**95.4**	**95.5**	**95.5(0.28)**	93.1	70.6	89.2(0.44)
IB	VIB	95.8	92.0	94.5(0.52)	94.7	94.8	94.8(0.33)	92.8	68.6	88.3(0.94)
VIB	VIB	95.8	92.2	94.6(0.32)	94.6	94.8	94.7(0.33)	92.8	70.1	88.5(0.68)
IB	IB	95.0	90.7	93.5(0.47)	94.5	94.6	94.6(0.48)	**93.2**	**71.1**	**89.5(0.60)**

实验结果可以看出，变分只在特定的情况下产生效果。分析表 5.4，在将变分操作应用于实体通道时，模型性能较差，说明变分操作并不适合用于实体嵌入；而将变分运算应用于上下文嵌入时，在大多数情况下，结果要优于普通瓶颈模块。

综上所述，VIB 网络对于上下文表示的扰动更加鲁棒，促使我们采用 VIB+IB 作为最终模型体系结构。另外，BERT 仅仅使用了一个简单的均值池化操作（mean pooling）来构建上下文表示，在提取上下文特征时，BERT 不可避免地集成了噪声，VIB 可以将这些噪声从最终的表示中去除。

（5）跨数据集模型效果验证

表 5.5 展示了跨数据集实验的结果，第一列中的 X→Y 指在 X 训练集上训练的模型在 Y 测试集上验证，S、R、C 分别表示 SemEval、ReLocaR 和 CoNLL 数据集。由于标注方式和标签分布的差异，在 SemEval 和 ReLocaR 上训练的模型泛化效果并不理想。相比之下，因为 CoNLL 包含的数据比其他数据集多几个数量级，使得模型更容易泛化，所以在 CoNLL 上训练的模型在 ReLocaR 和 SemEval 上验证的效果都不差。此外，在 ReLocaR 和 SemEval 上训练的模型在 CoNLL 上验证所得性能要优于 ReLocaR 和 SemEval，这也支持了我们的猜想，即更全面、更高质量的训练数据集对于转喻识别的最终效果提升巨大。

从模型的角度分析，由于缺少实体信息，BERT+MARK 在三个模型中表现最差。在整个跨域的实验中，MIXINFBERT 对域外数据集进行了更好的泛化，所以 MIXINFBERT 在该实验中取得了更好的表现。

表 5.5 跨数据集模型效果验证

源→目标	BERT+MARK			EBERT			MIXINFBERT		
	F1-L	F1-M	Acc(std)	F1-L	F1-M	Acc(std)	F1-L	F1-M	Acc(std)
R → S	80.5	59.2	73.7(1.56)	**81.1**	**60.0**	**74.4(1.15)**	80.8	59.9	74.1(2.01)
R → C	88.5	82.2	86.1(2.62)	88.9	82.8	86.5(1.88)	**89.9**	**83.6**	**87.6(1.39)**
C → R	92.9	92.5	92.7(0.58)	**93.2**	**93.0**	**93.2(0.59)**	93.0	92.4	92.7(0.71)
C → S	88.5	66.9	83.0(1.20)	88.7	54.4	82.5(1.95)	**88.8**	**66.1**	**83.0(1.30)**
S → R	75.3	56.4	68.5(1.89)	80.1	69.6	76.0(1.60)	**80.3**	**70.2**	**76.4(0.90)**
S → C	86.1	65.9	80.3(1.95)	89.5	75.0	85.2(1.04)	**88.9**	**73.9**	**84.4(0..72)**

（6）不同超参数下模型效果对比

表 5.6 不同超参数下模型效果对比（SemEval）

z_dim	β=1.0			β=0.1			β=0.01			β=0.001		
	F1-L	F1-M	Acc	F1-L	F1-M	Acc	F1-L	F1-M	Acc	F1-L	F1-M	Acc
32	92.8	68.9	88.3	92.8	69.5	88.4	92.3	69.1	88.1	93.2	70.3	88.9
64	93.1	70.2	88.8	92.9	68.5	88.4	91.8	65.7	87.9	93.1	70.3	88.8
128	93.4	**73.1**	89.4	93.1	69.2	88.8	93.2	69.4	88.9	**93.5**	72.9	**89.5**
256	93.1	70.2	88.8	93.1	71.4	88.9	93.1	70.6	89.2	93.1	71.3	88.9

表 5.7 不同超参数下模型效果对比（ReLocaR）

z_dim	β=1.0			β=0.1			β=0.01			β=0.001		
	F1-L	F1-M	Acc	F1-L	F1-M	Acc	F1-L	F1-M	Acc	F1-L	F1-M	Acc
32	94.6	94.7	94.7	94.2	94.3	94.3	95.0	95.1	95.1	94.7	94.9	94.8
64	95.2	95.2	95.2	95.3	**95.6**	**95.5**	94.8	95.0	94.9	94.4	94.6	94.6
128	95.0	95.1	95.1	94.6	94.7	94.7	95.2	95.3	95.3	94.7	94.9	94.8
256	95.0	95.1	95.1	94.5	94.4	94.3	**95.4**	95.5	95.5	95.0	95.3	95.2

超参数的选择（例如z_dim和β）会影响模型的性能。表 5.6 和表 5.7 中分别给出了在 SemEval 和 ReLocaR 数据集上超参数变化时的模型的结果。z_dim为嵌入表示 z的维度大小，当z_dim=128 或 256 时结果更好，说明嵌入表示维度过小可能会导致实体语义表示不全；β为模型的损失函数系数，在多数数据集中，当β=0.01 时，模型的性能达到最佳。

5.5.2 错误分析

本节使用几个典型的例子从多个角度分析 BERT+MARK、EBERT 和 MIXINFBERT 的优势和劣势。

表中粗体表示的为例句的实体。“分类”为目标实体的类别标签，其次“√”和“×”表示三个模型判断的正确与否。

（1）多个实体

基于 BERT 的模型依赖于上下文来预测实体的转喻类别标签，然而，当一个句子包含多个实体时，模型预测往往会受到其他共现实体的干扰。S1、S2 展示了目标实体与其他实体同时出现的示例，在这样的文本中，BERT+MARK 同时预测错了 S1 和 S2；EBERT 正确预测了 S1，说明对于实体的重点关注可以在一定程度上解决共现问题，然而，EBERT 过于关注实体信息，导致了 S2 的预测错误；MIXINFBERT 平衡考虑上下文和实体信息，并进行去噪和泛化，正确预测了 S2 的分类。

表 5.8 语义模型错误分析

#	例句	分类	BERT+ MARK	EBERT	MIXINF
S1	尽管他习惯用“英格兰”而不是“**大不列颠**”这个词，但无论从血统上还是从性情上来说，他都很难相信自己是英国人。	转喻	×	√	√
S2	西太平洋的两个小型岛屿共和国，**瑙鲁**和基里巴斯，将带着提案抵达会场。	转喻	×	×	√
S3	伊拉克应**伊朗**要求将霍梅尼驱逐。	转喻	×	√	×
S4	**德国**的罗马风作品质量很高，而哥特式作品则较为局限。	非转喻	×	×	√

（2）所属关系

在例子中使用“的”表示实体之间的所属关系。句子 S4 中，由于“德国”和“罗马风作品”之间的所属关系干扰，BERT+MARK 未能识别出其分类为非转喻，而 MIXINFBERT 成功识别出了“德国”的非转喻含义。

在训练数据集中有很多表示所属关系的转喻文本，导致模型在某一类别上过拟合，MIXINFBERT 使用了一个变分正则化器，学习通过采样生成中间数据表示以挖掘深层次的信息，而不是仅仅依靠文本的表面含义。

（3）词汇误导

在 S4 中，MIXINFBERT 会被词汇“应”误导，而忽略了实体“伊朗”后面的重要词汇“要求”。在某些情况下，与传统模型相比，MIXINFBERT 在识别语义中的潜在陷阱方面还不够鲁棒，所以有必要在未来继续改进 MIXINFBERT，更多在转喻文本中挖掘强关联的线索，降低模型对特殊句式的敏感度。

5.6 本章小结

本章对如何实现转喻文本的深度语义表达进行了研究。针对信息冗余及上下文语言表述多样的问题，提出了一种基于变分信息瓶颈的语义压缩模型，该模型利用变分信息挖掘较为抽象的语义框架，减弱了转喻文本中不同的句式形态引起的误差，保证了转喻识别的高准确率。其中，变分正则化器弥补了模型高度依赖训练数据的缺点，防止过拟合，从而实现对转喻句子的深度理解，为转喻识别效率的提升提供了技术保障；而信息瓶颈压缩器使用信息压缩技术，去除转喻语句中的冗余信息，实现语义知识的泛化和去噪。最终，所提出的模型能够有效地解决转喻识别的深度语义表达问题，在数据量较少，数据质量较差的情况下，高质量地识别转喻。

第六章　结论与展望

6.1　结论

随着互联网信息技术的蓬勃发展和国民经济水平的快速提升，人工智能进入新时代，由机器感知上升到机器认知高度，如何自动、准确地让机器理解实体真实含义成为人们关注的焦点。在当前时代下，转喻仍是一种较难处理的实体指代含义识别任务，现有的基于深度学习的自然语言处理方法仅仅使用文本数据本身，可以实现对表面实体进行识别，但由于转喻语句隐含性高的特点，难以理解其背后蕴含的真实实体语义，无法实现机器认知。解决文本中的转喻识别问题，不仅可以拓宽信息获取的途径，还能够提升信息获取效率，降低人工成本，具有非常重要的研究意义和应用价值。

高效地实现转喻识别，需要理解和构建实体词与上下文之间的交互，并找到转喻词与上下文之间存在的语义冲突。其中，实体表示的准确与否关系到模型的整体性能，而上下文知识可以分为句法结构知识和语义信息知识，上下文信息的去噪能力和表示能力也决定了模型的最终效果。因此，本文从认知语言学角度出发，分别采用词汇、句法、语义层面的方法，研究转喻识别中的难点问题，具体贡献如下：

贡献一：解决了文本中实体含义无法精确表示的问题：一个完整的实体表示可以更好地帮助模型的理解，从而提供快速精准的识别反馈。目前的转喻识别模型仅仅依靠单一的上下文知识，准确率有限，因此，我们在转喻模型中融入了实体知识，解决了两个难点问题：使用实体指示器增强了实体词汇信息在转喻识别任务中的表达能力；构建一种融合实体词汇感知的上下文表示结构模型 EBERT，使得联合表示可以清晰有效地表达语义的冲突。

贡献二：解决了复杂句式中的实体真实语义问题：目前的转喻识别方法大多建立在语义信息的基础上，而对于长难句识别的困难，说明了转喻过程应既依赖于词汇和上下文语义，也依赖于句法知识。所以，我们需要句法结构辅助转喻的判断，在转喻识别中融入句法约束的过程中，解决了两个难点问题：使

用 GCN 完成句法融入，把句法知识完整高效地输入到模型中；构建了一种句法依赖权重分配方法 AGCN，有效地清除句法知识中的噪声。

贡献三：解决了文本上下文信息冗余且表述多样的问题：由于转喻识别是一个刚刚起步的方向，在转喻文本中，信息冗余、表达形式多样的现象非常常见，造成语义信息稀疏，对语义的理解变得困难，这样训练出来的模型更倾向于从高发生率的已知实例去进行判断，造成大量的错误预测。为了更完整地挖掘语句的深层信息，我们解决了两个难点问题：创建一种变分正则化方法有效处理未知文本数据，解决过拟合的问题；使用信息瓶颈语义压缩技术，捕捉固定的语义模式，解决语义信息冗余，实现文本去噪和泛化。

从本文的章节安排、联系来看，本文的工作总结如下：

第一章介绍了转喻识别的研究背景和意义，在现有研究的基础上提出了三个研究问题及相对应的研究内容，最后总结了本论文的创新点。

第二章介绍了当前转喻识别的国内外研究现状和相关概念及技术。

第三章提出了融合实体词汇感知的上下文表示方法和模型。该模型以预训练模型 BERT 作为基线模型，并添加了强大的实体感知能力；针对实体表达能力弱的缺点，模型利用实体指示器，指示实体在句子中的语义和位置信息，增强实体词汇信息在转喻识别中的表达能力；针对实体信息和上下文信息如何高效融合的问题，构建了一种融合实体词汇感知的上下文表示结构清晰有效地表达二者的联合语义。实验结果表明，我们的模型在中英转喻识别数据集上的效果相较于单纯的预训练模型有较大的提高，同时举例说明了实体信息对于转喻识别的重要作用，证明了实体融入方法的有效性。

第四章提出了一种基于注意力指导的图卷积网络句法约束模型。该方法将句法知识融入图卷积网络中，训练句子上下文语义和句法依赖的图表示；同时，通过注意力机制对依存关系分配相应权重，不仅突出了依赖中关键信息的表达，提升了模型抵御句法噪声干扰的能力，提高模型的准确率和召回率，而且增加了句法知识融入的可解释性。实验结果表明，对比已有模型，我们的模型有着更好的准确率和 F1 值，同时，对于长难句的识别效果更好，弥补了 BERT 模型

的缺陷。另外，我们通过注意力矩阵可视化实验，对比加入权重分配模块前后的注意力矩阵，证明了句法约束方法的有效性。

第五章提出了一种基于变分信息瓶颈的语义压缩技术模型。该模型使用变分信息弥补了模型高度依赖训练数据的缺点，防止过拟合，实现了对转喻句子的深度理解；并使用了信息瓶颈压缩技术，去除转喻语句中的冗余信息，实现语义知识的泛化和去噪，从而提升语义表示的准确率，为转喻识别效率的提升提供了技术保障。实验结果表明，我们的模型得到了比已有模型更好的结果。同时，我们进行了语义压缩模块和变分正则化模块的消融实验，证明了语义压缩方法的有效性。

6.2 展望

本文对转喻识别的实体表示、句法约束以及语义表达这三个重要问题进行了相关研究。通过实验验证与分析，我们提出的方法取得了较好的效果，但是在以下方面仍能有一些改进和深入的工作：

（1） 实体上下文表示方法的改进

本文在第三章中提出了一种融合实体词汇感知的上下文表示方法，通过实体指示器和实体上下文联合表示，提升了实体在表示中的表达能力，该方法的有效性也在实验中得到了验证。但是，该方法对与实体表示的构建方法直接使用了平均池化算法，没有对每个 token 表示中的重要特征进行集中提取。因此，接下来可以对该提取过程进行优化，使用权重分配策略来提升实体表示的精确性。

（2）依存关系融入的改进

本文在第四章中提出了一种基于注意力指导的图卷积网络句法约束模型，通过加入词与词之间的依存关系实现了句法知识的融合表示。但是，该方法仅对每一个 token 的依存关系存在与否进行区分，而没有加入依存关系的种类标签，如主谓关系等，所以句法知识并没有得到完全的利用，接下来可以对依存关系的提取过程进行优化，来确保依存融入的完整性。

（3）构建外部知识库

本文在第五章中提出了基于变分信息瓶颈的语义压缩技术模型，提高了转喻句子语义表达的准确性。我们的方法可以比较清晰地提取句子中已有的语义知识，然而，在很多情况下，依然需要融入外部知识帮助识别转喻，以填补句子中语义的缺失。接下来可以尝试构建外部知识库，并进行实体链接来联系实体和知识库中的外部知识，进一步提升文本语义的丰富性。

参考文献

【1】. 田嘉, 苏畅, 陈怡疆 . 隐喻计算研究进展[J]. 软件学报, 2015, 26(1): 40-51.

【2】. Graves A. Long Short-Term Memory[M]//Supervised Sequence Labelling with Recurrent Neural Networks. Springer, Berlin, Heidelberg, 2012: 37-45.

【3】. Peters M, Neumann M, Iyyer M, et al. Deep Contextualized Word Representations [C]//Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers). 2018: 2227-2237.

【4】. Devlin J, Chang M W, Lee K, et al. Bert: Pre-Training of Deep Bidirectional Transformers for Language Understanding[C]//Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers). 2019: 4171-4186.

【5】. 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.

【6】. Mao R., Lin C., Guerin F. End-to-End Sequential Metaphor Identification Inspired by Linguistic Theories[C]//Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics. 2019: 3888–3898.

【7】. Gritta M., Pilehvar M. T., Limsopatham N., et al. Vancouver welcomes you! minimalist location metonymy resolution[C]//Proceedings of Annual Meeting of the Association for Computational Linguistics, volume 1. 2017: 1248–1259.

【8】. Li H., Vasardani M., Tomko M., et al. Target word masking for location metonymy resolution[C]//Proceedings of the 28th International Conference on Computational Linguistics. 2020: 3696–3707.

【9】. Pustejovsky J. The generative lexicon[M]//Computational Linguistics, 1991: 17(4):409–441.

【10】. Chan Y S, Roth D. Exploiting syntactico-semantic structures for relation

extraction[C]//Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies. Portland, Oregon, USA. Association for Computational Linguistics, 2011: 551–560.

【11】. Fundel K, Kuffner R, Zimmer R. RelEx—relation extraction using dependency parse trees[J]. Bioinformatics, 2007, 23(3): 365–371.

【12】. Janda L A. Metonymy in word-formation[J]. Cognitive Linguistics, 2011, 22(2):359–392.

【13】. Nissim M, Markert K. Syntactic features and word similarity for supervised metonymy resolution[C]//Proceedings of the 41st Annual Meeting of the Association for Computational Linguistics. 2003: 56–63.

【14】. Nastase V, Strube M. Combining collocations, lexical and encyclopedic knowledge for metonymy resolution[C]//Proceedings of the conference on Empirical Methods in Natural Language Processing. 2009: 910–918.

【15】. Nastase V, Judea A, Markert, K, et al. Local and global context for supervised and unsupervised metonymy resolution[C]//Proceedings of the 2012 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning. 2012: 183–193.

【16】. Zhang Y, Qi P, Manning C D. Graph convolution over pruned dependency trees improves relation extraction[C]//Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing. 2018: 2205–2215.

【17】. Guo Z, Zhang Y, Lu W. Attention guided graph convolutional networks for relation extraction[C]//Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics. 2019: 241–251.

【18】. Mahabadi R K, Belinkov Y, and Henderson J. Variational Information Bottleneck for Effective Low-Resource Fine-Tuning[C]//Proceedings of the International Conference on Learning Representations. 2020.

【19】. Lakoff G. Image metaphors[J]. Metaphor and Symbol, 1987, 2(3):219–222.

【20】. Kim Y. Convolutional Neural Networks for Sentence Classification[C]//Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP).

2014: 1746-1751.

【21】. Seo M, Kembhavi A, Farhadi A, et al. Bidirectional Attention Flow for Machine Comprehension[C]//Proceedings of the 5th International Conference on Learning Representations. 2017.

【22】. Lakoff, G. Metaphor and war: The metaphor system used to justify war in the gulf[J]. Peace Research, 1991, 25–32.

【23】. Zhang M, Ai X, Hu Y. Chinese Text Classification System on Regulatory Information Based on SVM[C]//IOP Conference Series: Earth and Environmental Science. IOP Publishing, 2019, 252(2): 022133.

【24】. Markert K, Nissim M. Semeval-2007 task 08: Metonymy resolution at semeval-2007[C]//Proceedings of the 4th International Workshop on Semantic Evaluations. 2007: 36–41.

【25】. Farkas R, Simon E, Szarvas, G, et al. Gyder: Maxent metonymy resolution[C]//Proceedings of the 4th International Workshop on Semantic Evaluations. 2007: 161–164.

【26】. Brun C, Ehrmann M, Jacquet G. XRCE-M: A hybrid system for named entity metonymy resolution[C]//Proceedings of the Fourth International Workshop on Semantic Evaluations (SemEval-2007). 2007: 488–491.

【27】. Nastase V, Strube M. Transforming wikipedia into a large scale multilingual concept network[J]. Artificial Intelligence, 2013, 194:62–85.

【28】. Yang S, Feng D, Qiao L, et al. Exploring Pre-trained Language Models for Event Extraction and Generation[C]//Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics. Association for Computational Linguistics, 2019: 5284–5294.

【29】. Mathews K A, Strube M. A large harvested corpus of location metonymy[C]//Proceedings of the 12th Language Resources and Evaluation Conference. Marseille, France. European Language Resources Association, 2020: 5678–5687.

【30】. Socher R, Perelygin A, Wu J, et al. Recursive deep models for semantic compositionality

over a sentiment treebank[C]//Proceedings of the 2013 conference on empirical methods in natural language processing. 2013: 1631–1642.

【31】. Rajpurkar P, Zhang J, Lopyrev K, et al. SQuAD: 100,000+ questions for machine comprehension of text[C]//Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing. 2016: 2383–2392.

【32】. Glavaš G, Vulić I. Is supervised syntactic parsing beneficial for language understanding tasks? an empirical investigation[C]//Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume. Association for Computational Linguistics. 2021: 3090–3104.

【33】. 赵红艳.基于机器学习与语义知识的动词隐喻识别[J]. 南京师范大学学报，2011，11(3):59-64

【34】. 贾玉祥，咎红英，范明等.面向隐喻识别的词语抽象性度量[J]. 中文信息学报，2017, 31(3): 41-47.

【35】. 游维，周昌乐.基于统计的汉语隐喻生成模型及其系统实现[J]. 心智与计算，2007, (1): 133-141.

【36】. 苏畅，周昌乐.基于合作机制的汉语名词性隐喻理解方法[J]. 计算机应用研究，2007, 24(9): 67-70.

【37】. Vaswani A, Shazeer N, Parmar N, et al. Attention Is All You Need[C]//Advances in Neural Information Processing Systems. 2017: 5998-6008.

【38】. Yang Z, Dai Z, Yang Y, et al. Xlnet: Generalized autoregressive pretraining for language understanding[J]. Advances in neural information processing systems, 2019.

【39】. Robinson, Jane J. Dependency structures and transformational rules[J]. Language, 1970: 259-285.

【40】. Mikolov T, Sutskever I, Chen K, et al. Distributed Representations of Words and Phrases and Their Compositionality[C]//Advances in Neural Information Processing Systems. 2013: 3111-3119.

【41】. Pennington J, Socher R, Manning C D. Glove: Global Vectors for Word Representation[C]//Proceedings of the 2014 Conference on Empirical Methods in Natural

Language Processing (EMNLP). 2014: 1532-1543.

【42】. Melamud O, Goldberger J, Dagan I. Context2vec: Learning Generic Context Embedding with Bidirectional LSTM[C]//Proceedings of the 20th SIGNLL Conference on Computational Natural Language Learning. 2016: 51-61.

【43】. Mesnil, G., He, X., Deng, L., and Bengio, Y. Investigation of recurrent-neural-network architectures and learning methods for spoken language understanding[C] //INTERSPEECH 2013, 14th Annual Conference of the International Speech Communication Association. Lyon, France, August 25-29, 2013: 3771–3775.

【44】. Baroni M, Dinu G, Kruszewski G. Don't count, predict! a systematic comparison of context-counting vs. context-predicting semantic vectors[C]//Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers). 2014: 238–247.

【45】. Collobert R, Weston J, Bottou L, et al. Natural language processing (almost) from scratch[J]. Journal of Machine Learning Research, 2011, 12: 2493–2537.

【46】. Yang S, Feng D, Qiao L, et al. Exploring pre-trained language models for event extraction and generation[C]//Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics. 2019: 5284-5294.

【47】. Wu S, He Y. Enriching pre-trained language model with entity information for relation classification[C]//Proceedings of the 28th ACM International Conference on Information and Knowledge Management. 2019: 2361–2364.

【48】. Li D, Wei F, Tan C, et al. Adaptive recursive neural network for target-dependent twitter sentiment classification[C]//Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers). 2014: 49–54.

【49】. Peng N, Poon H, Quirk C, et al. Cross-sentence n-ary relation extraction with graph LSTMs[J]. Transactions of the Association for Computational Linguistics, 2017, 5: 101–115.

【50】. Liu J, Chen Y, Liu K, et al. Event extraction as machine reading comprehension[C]//Proceedings of the 2020 Conference on Empirical Methods in Natural

Language Processing (EMNLP). 2020: 1641-1651.

【51】. Joshi M, Penstein-Rosé C. Generalizing dependency features for opinion mining[C]//Proceedings of the ACL IJCNLP 2009 Conference Short Papers. 2009: 313–316.

【52】. 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.

【53】. Kambhatla N. Combining lexical, syntactic, and semantic features with maximum entropy models for extracting relations[C]//Proceedings of the ACL 2004 on Interactive poster and demonstration sessions, 22. Association for Computational Linguistics, 2004.

【54】. Zhang M, Zhang J, Su J. Exploring syntactic features for relation extraction using a convolution tree kernel[C]//Proceedings of the Human Language Technology Conference of the NAACL, Main Conference. Association for Computational Linguistics, 2006: 288–295.

【55】. Xu Y, Mou L, Li G, et al. Classifying relations via long short term memory networks along shortest dependency paths[C]//Proceedings of the 2015 conference on Empirical Methods in natural language processing. 2015: 1785–1794.

【56】. Liu Y, Wei F, Li S, et al. A dependency-based neural network for relation classification[C]//Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 2: Short Papers), 2015: 285–290.

【57】. 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.

【58】. Jawahar G, Sagot B, Seddah D. What Does BERT Learn about the Structure of Language?[C]//Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics. 2019: 3651–3657.

【59】. Bahdanau D, Cho K, Bengio Y. Neural machine translation by jointly learning to align

and translate[C]//Proceedings of the 3rd International Conference on Learning Representations. 2015.

【60】. Luong T, Pham H, Manning C D. Effective approaches to attention-based neural machine translation[C]//Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing, Lisbon, Portugal. Association for Computational Linguistics, 2015: 1412–1421.

【61】. Huang G, Liu Z, Van Der Maaten L, et al. Densely connected convolutional networks[C]//Proceedings of the IEEE conference on computer vision and pattern recognition. 2017: 4700–4708.

【62】. Manning C D, Surdeanu M, Bauer J, et al. The Stanford CoreNLP natural language processing toolkit[C]//Proceedings of 52nd Annual Meeting of the Association for Computational Linguistics: System Demonstrations. 2014: 55–60.

【63】. Sundermeyer M, Schlüter R, Ney H. LSTM neural networks for language modeling[C]// Thirteenth annual conference of the international speech communication association. 2012.

【64】. Si C, Chen W, Wang W. An attention enhanced graph convolutional LSTM network for skeleton-based action recognition[C]//Proceedings of the IEEE conference on computer vision and pattern recognition. 2019: 1227–1236.

【65】. Hochreiter S, Schmidhuber J. Long short-term memory[J]. Neural computation, 1997. 9(8): 1735–1780.

【66】. Tang G, Muller M, Gonzales A R, et al. Why self-attention? a targeted evaluation of neural machine translation architectures[C]//Empirical Methods in Natural Language Processing. 2018: 4263–4272.

【67】. Shibata T, Kawahara D, Kurohashi S. Neural network-based model for Japanese predicate argument structure analysis[C]//Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers). 2016: 1235–1244.

【68】. Kingma D P, Welling M. Auto-encoding variational bayes[C]//In International Conference on Learning Representations. 2014.

【69】. Danilo Jimenez Rezende, Shakir Mohamed, and Daan Wierstra. 2014. Stochastic Backpropagation and Approximate Inference in Deep Generative Models. In Proceedings of the 31st International Conference on Machine Learning (ICML).

【70】. Chen X, Duan Y, Houthooft R, et al. InfoGAN: Interpretable Representation Learning by Information Maximizing Generative Adversarial Nets[C]//Proceedings of the 30th International Conference on Neural Information Processing Systems. 2016: 2180-2188.

【71】. Alemi A A, Fischer I, Dillon J V, et al. Deep Variational Information Bottleneck[C]//Proceedings of the 2017 International Conference on Learning Representations (ICLR). 2017.

【72】. Lakoff G, Johnson M. Conceptual metaphor in everyday language[J]. The Journal of Philosophy, 1980, 77(8): 453–486.

【73】. Lakoff G. The contemporary theory of metaphor[M]//Metaphor and Thought (2nd edition), 1993.

【74】. Fass D. Metonymy and metaphor: What's the difference?[C]//Coling Budapest 1988 Volume 1: International Conference on Computational Linguistics. 1988.

【75】. Sun C, Huang L, Qiu X. Utilizing BERT for aspect-based sentiment analysis via constructing auxiliary sentence[C]//Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers), 2019: 380–385.

【76】. Lin C, Miller T, Dligach D, et al. A BERT-based universal model for both within and cross-sentence clinical temporal relation extraction[C]//Proceedings of the 2nd Clinical Natural Language Processing Workshop. 2019: 65–71.

【77】. Qu C, Yang L, Qiu M, et al. Bert with history answer embedding for conversational question answering[C]//Proceedings of the 42nd International ACM SIGIR Conference on Research and Development in Information Retrieval, 2019: 1133–1136.

【78】. Brickley D, Burgess M, Noy N. Google Dataset Search: Building a search engine for datasets in an open Web ecosystem[C]//The World Wide Web Conference. 2019: 1365-1375.

【79】. Nyagadza B. Search engine marketing and social media marketing predictive trends[J]. Journal of Digital Media & Policy, 2021.

【80】. Raganato A, Tiedemann J. An analysis of encoder representations in transformer-based machine translation[C]//Proceedings of the 2018 EMNLP Workshop BlackboxNLP: Analyzing and Interpreting Neural Networks for NLP. The Association for Computational Linguistics, 2018.

【81】. Zhou J, Cao Y, Wang X, et al. Deep Recurrent Models with Fast-Forward Connections for Neural Machine Translation[J]. Transactions of the Association for Computational Linguistics, 2016, 4: 371-383.

【82】. Clark J H, Choi E, Collins M, et al. TyDi QA: A benchmark for information-seeking question answering in typologically diverse languages[J]. Transactions of the Association for Computational Linguistics, 2020, 8: 454-470.

【83】. Vakulenko S, Longpre S, Tu Z, et al. Question rewriting for conversational question answering[C]//Proceedings of the 14th ACM International Conference on Web Search and Data Mining. 2021: 355-363.

作者在攻读硕士学位期间公开发表的论文

【1】. XXX. An Extensible Framework of Leveraging Syntactic Skeleton for Semantic Relation Classification[J]. ACM Trans. on Asian Low Resource Language Information Processing (TALLIP), 2020, 19(6): 77:1-77:21. （第三作者）

【2】. XXX. An Empirical Study of Incorporating Syntactic Constraints into Bert-based Location Metonymy Resolution[J]. Natural Lanaguge Engineering (NLE), （审稿中，终审，第二作者，导师一作）

【3】. XXX. Mixed Information Bottleneck for Location Metonymy Resolution Using Pre-trained Language Models[J]. ACM Trans. on Asian Low Resource Language Information Processing (TALLIP). （审稿中，第二作者，导师一作）

作者在攻读硕士学位期间所参与的项目

1. 项目来源：2021 年度上海市“科技创新行动计划”扬帆计划项目

   项目名称：面向药物警戒的跨语言多知识驱动不良事件挖掘方法

   项目编号：XXX

   执行期限：2021～2024

致　　谢

光阴荏苒，时光飞逝，一转眼就到了毕业的时节，回想起踏入研究生校园的那一刻，仿佛就在眼前。

在三年的学习生活中，王昊老师给了我细心的学术指导以及良好的学习环境，使我从中获益匪浅。王老师勤勉的作风，负责的态度，永远留在了我心里。在此，谨向王老师致以深深的敬意和由衷的感谢！

另外，我还要感谢实验室的老师和同学们，在我的学习和生活中带来了很多照顾，在这里诚挚的向骆祥峰老师的谆谆教诲表示感谢，同时也向高剑奇师兄、赵丹阳师姐、金伟强同学、王子健同学等表达谢意，是你们的关怀和帮助让我度过了艰苦但充满意义的研究生生涯！

同时我还要感谢我的父母，是他们给予内向敏感的我坚持学习的信心和力量，在我承受了较大的学业压力的情况下，他们依然孜孜不倦地开导我，鼓励我，在物质上和精神上给了我莫大的帮助！

象牙塔中的日子即将结束，我将怀着激动且忐忑的心情第一次踏上社会，不管前路有多少风雨，我相信我的同学们、朋友们、老师们和亲人们会永远站在我的身后，坚定地鼓励和支持我，最后，再一次向你们表示感谢！