

融合富信息的英文文本自动校对

(申请清华大学工学博士学位论文)

培 养 单 位： 计 算 机 科 学 与 技 术 系

学 科： 计 算 机 科 学 与 技 术

研 究 生： 刘 正 皓

指 导 教 师： 孙 茂 松 教 授

二〇二一年六月

Automatic Proofreading of English Text with Rich Information

Dissertation Submitted to

Tsinghua University

in partial fulfillment of the requirement

for the degree of

Doctor of Science

in

Computer Science and Technology

by

Liu Zhenghao

Dissertation Supervisor: Professor Sun Maosong

June, 2021

学位论文公开评阅人和答辩委员会名单

公开评阅人名单

马少平	教授	清华大学
刘洋	教授	清华大学

答辩委员会名单

主席	孙乐	研究员	中国科学院软件研究所
委员	孙茂松	教授	清华大学
	周明	研究员	创新工场
	刘洋	教授	清华大学
	黄民烈	教授	清华大学
	丁贵广	副教授	清华大学
秘书	陈慧敏	助理研究员	清华大学

关于学位论文使用授权的说明

本人完全了解清华大学有关保留、使用学位论文的规定，即：

清华大学拥有在著作权法规定范围内学位论文的使用权，其中包括：(1) 已获学位的研究生必须按学校规定提交学位论文，学校可以采用影印、缩印或其他复制手段保存研究生上交的学位论文；(2) 为教学和科研目的，学校可以将公开的学位论文作为资料在图书馆、资料室等场所供校内师生阅读，或在校园网上供校内师生浏览部分内容；(3) 根据《中华人民共和国学位条例暂行实施办法》及上级教育主管部门具体要求，向国家图书馆报送相应的学位论文。

本人保证遵守上述规定。

(保密的论文在解密后遵守此规定)

作者签名：_____

导师签名：_____

日 期：_____

日 期：_____

摘要

文本校对是文章出版的重要程序，其可以为个人、企业以及政府部门提供文本审查服务，从而保证了发布信息的语法以及语义层面的准确性、真实性，能够防止错误信息的传播。然而，文本校对是一个细致的工作，人工校对往往存在着漏查、效率低、速度慢等问题，因此如何能够自动化地对文本进行语法层面以及语义层面的校对是一个重要研究问题。

本文针对英文文本自动校对中的两个核心任务，语法改错任务以及事实验证任务，进行了研究并利用预训练语言模型实现了对文本的自动化校对。本文通过融合诸如语言知识、世界知识以及领域知识等富信息，进一步辅助文本校对模型对文本的语法以及事实层面的错误进行检查。本文针对融合富信息的文本自动校对任务中的缺点和不足，系统性地进行了如下四个工作。

针对在英文语法错误检查中缺乏语法校对证据的问题，本文首先讨论了**语法改错算法的框架和语法改错证据获取方法**。该工作对比了通用语言模型预训练方法以及不同的面向语法改错场景的语言模型预训练策略，确定了对语法改错模型最优的预训练策略。此外，该工作通过对包含噪音的训练语料库进行筛选，进一步提升了语法改错模型的性能。最后，本工作利用训练好的语法改错模型通过柱搜索解码为语法错误检查模型提供若干个可能的语法改错结果。这些语法改错结果对可能出现语法错误的位置起到提示作用，进一步辅助语法错误检查模型。

为了能够融合来自语法改错模型、世界知识库以及领域知识库的文本校对证据来辅助文本进行错误检查，本文针对于语法错误检查以及事实验证两个任务，分别提出了**融合多语法改错证据的英文语法校对模型**以及**融合多事实证据的细粒度联合推理的事实验证模型**。两个模型分别考虑了语法错误检查以及事实验证任务的特点，通过不同的证据抽取机制得到语法和语义校对线索，进而辅助预训练语言模型进行文本错误检查。此外我们的语法错误检查模型可以进一步通过对语法改错结果进行质量评估来提升语法改错效果。

为了解决在专业领域下的文本事实验证问题，本文提出了**面向专业领域的事实验证模型**，以此来增强语言模型对专业领域中文本的理解能力。该方法设计了两种不同的语言模型继续训练机制，从而帮助语言模型学习专业领域中专有名词的语义信息，进一步提升了事实验证模型在面向专业领域的事实验证效果。

关键词：文本校对；富信息；语法改错；语法错误检查；事实验证

Abstract

Text proofreading is an important procedure for article publishing. It can provide text reviews for individuals, enterprises, and government departments, ensuring the accuracy and authenticity of the grammatical and semantic correctness of the published articles and preventing the spread of misinformation. However, text proofreading is meticulous work, and manual proofreading often faces some problems, such as omission and low efficiency. Hence how to automatically proofread texts at the grammatical and semantic levels is an important research problem in the NLP community.

This work aims at the two core tasks, grammatical error correction, and fact verification, to automatically proofread English texts with pre-trained language models. This article integrates rich information, such as language knowledge, world knowledge, and specific domain knowledge, to further assist the text proofreading model to check the grammatical and factual errors in the text. To solve the problems in automatic proofreading of English text with rich information, this work systematically carries out the following four studies.

This article first leverages the grammatical error correction models to generate grammatical error correction evidence for grammatical error detection models. This work compares the general language model pre-training methods and different pre-training strategies for grammatical error correction. Then this work determines the optimal pre-training strategy for grammatical error correction models. Besides, this work further trains grammatical error correction models by filtering the training corpus that contains noise to further improve model performance. Finally, this work uses the well-trained grammatical error correction model to provide several grammatical error correction results for the grammatical error detection model via beam search decoding to annotate the potential grammatical errors and assist grammatical error detection models.

To integrate the text proofreading evidence from the grammatical error correction model, the world knowledge base, and the knowledge base of a specific domain, this paper proposes two models to fuse multiple proofreading evidence for the text error detection, **the grammatical error detection model with multiple grammatical error correction results** and **the fine-grained fact verification model with multi-evidence reasoning** to assist the two proofreading tasks, grammatical error detection, and fact verification.

These two models consider the characteristics of proofreading at the grammatical level and the semantic level and design different methods to extract proofreading clues from rich information that can assist the pre-trained language models in text error detection tasks. Besides, our grammatical error detection model can further improve grammatical error correction models through quality estimation.

To solve the problem of fact verification in the specific domain, this paper proposes the enhanced pre-trained language model to improve its language modeling ability and text reasoning ability in the specific domain. This method proposes two different continuous training strategies that train language models on the data of the specific domain to help language models learn the word semantics in the specific domain and improve the fact verification performance in the specific domain.

Keywords: Text Proofreading; Rich Information; Grammatical Error Correction; Grammatical Error Detection; Fact Verification

目 录

摘 要.....	I
Abstract.....	II
目 录.....	IV
插图清单.....	VII
附表清单.....	VIII
第 1 章 引言	1
1.1 研究背景和意义.....	1
1.2 英文文本校对任务	2
1.2.1 语法错误校对.....	2
1.2.2 事实错误校对.....	4
1.3 融合富信息的英文文本自动校对.....	6
1.4 本文的主要研究内容	8
第 2 章 语法改错算法的框架和语法改错证据获取	11
2.1 问题描述.....	11
2.2 相关工作.....	13
2.3 语法改错数据集.....	14
2.4 语法改错预训练策略	16
2.5 基于强化学习的噪音数据筛选方法	18
2.6 实验.....	19
2.6.1 实验设置.....	19
2.6.2 不同预训练方法在语法改错任务上的效果.....	20
2.6.3 英文学习者网站数据 Lang8 对于语法改错效果影响.....	22
2.6.4 通过筛选有噪音语料库提升语法改错效果.....	22
2.7 本章小结.....	24
第 3 章 融合多语法改错证据的英文语法校对	25
3.1 问题描述.....	25
3.2 相关工作.....	29

3.3	模型框架	30
3.3.1	句子对表示初始化	30
3.3.2	通过多语法改错结果验证词语质量	31
3.3.3	语法改错质量评估	33
3.3.4	端到端训练	34
3.4	实验	34
3.4.1	数据集	35
3.4.2	基线模型	36
3.4.3	实验设置	38
3.4.4	语法错误检查效果	39
3.4.5	语法改错质量评估效果	40
3.4.6	通过语法改错质量评估模型提升语法改错效果	42
3.4.7	融合多语法改错质量评估模型特征对语法改错效果影响	45
3.4.8	VERNet 模型中注意力机制样例分析	46
3.4.9	语法改错样例分析	47
3.5	本章小结	49
第 4 章	融合多事实证据的细粒度联合推理的事实验证	50
4.1	问题描述	50
4.2	相关工作	53
4.3	模型框架	55
4.3.1	基于推理图的多证据联合推理	55
4.3.2	推理图节点表示初始化	56
4.3.3	基于 Edge Kernel 的证据信息传播	57
4.3.4	基于 Node Kernel 的证据选择	59
4.4	实验	60
4.4.1	数据集	60
4.4.2	基线模型	61
4.4.3	实验设置	62
4.4.4	事实验证效果	63
4.4.5	不同推理场景下模型效果评估	65
4.4.6	核函数在事实验证中的作用	66
4.4.7	样例分析	70
4.5	本章小结	71

第 5 章 面向专业领域的事实验证	72
5.1 问题描述	72
5.2 相关工作	74
5.3 模型框架	75
5.3.1 事实验证流水线	75
5.3.2 面向专业领域的语言模型继续训练	77
5.4 实验	79
5.4.1 数据集	79
5.4.2 基线模型	80
5.4.3 实验设置	80
5.4.4 面向专业领域的事实验证流水线模型效果	81
5.4.5 语言模型在文档检索任务上的效果	82
5.4.6 语言模型在句子检索任务上的效果	83
5.4.7 语言模型在事实验证任务上的效果	84
5.4.8 样例分析	85
5.5 本章小结	86
第 6 章 总结与展望	87
6.1 主要贡献	87
6.2 未来工作展望	88
参考文献	90
致 谢	99
声 明	100
声 明	101
个人简历、在学期间完成的相关学术成果	102
指导教师学术评语	104
答辩委员会决议书	105

插图清单

图 1.1	一些商用文本校对工具。	1
图 1.2	语法错误检查任务以及语法纠错任务示意图。	3
图 1.3	事实验证流程示意图。	5
图 1.4	融合富信息的英文文本自动校对方法。	6
图 1.5	文本自动校对所面临的挑战。	7
图 1.6	本文研究内容框架。	9
图 2.1	语法纠错任务示意图。	12
图 2.2	基于强化学习的语法纠错语料筛选方法。	18
图 3.1	柱搜索解码所生成的语法纠错结果的语法合理性。	27
图 3.2	柱搜索解码所生成的语法纠错结果的语法纠错质量。	27
图 3.3	基于多语法纠错假设的神经验证网络 (Neural Verification Network, VER-Net) 模型框架图。	31
图 3.4	VERNet [†] 模型在 BEA19 数据上不同语法错误类型上的表现。	44
图 3.5	VERNet [†] 模型中不同特征权重分布。	45
图 3.6	VERNet 模型中两种注意力机制权重的可视化结果。	46
图 4.1	英文文本事实验证的一个例子。	51
图 4.2	KGAT 模型框架图。	55
图 4.3	KGAT 注意力权重熵值。	67
图 4.4	KGAT 和 GAT 的证据选择模块效果展示。	68
图 4.5	来自 GAT 和 KGAT 的证据传播时注意力机制对于证据中每个词语的注意力权重分布。	69
图 4.6	证据传播时的注意力机制 (Edge Attention) 对于证据中词语的注意力权重分配。	70
图 5.1	在新型冠状病毒肺炎有关的问题中单词被拆解成 Subwords 的单词比例。	73
图 5.2	面向专业领域的语言模型继续训练方法示意图。	77

附表清单

表 2.1	常见的一些语法错误示例。	14
表 2.2	语法纠错数据集信息统计。	15
表 2.3	由拼写检查器 Aspell 所提供的潜在的混淆词示例。	16
表 2.4	综合增广策略 ^[19] 在标注语料微调阶段所采用的参数。	20
表 2.5	不同预训练方法在语法纠错任务上的效果（单一模型比较）。	21
表 2.6	加入 Lang8 数据对不同语法纠错模型的影响。	21
表 2.7	不同 Lang8 数据筛选策略对语法纠错模型的效果。	22
表 2.8	基于强化学习的数据筛选方法筛选出数据在文本的修改操作类别上的数量统计。	22
表 2.9	基于强化学习的数据筛选方法筛选出数据所保留的语法错误类别数量统计。	23
表 3.1	语法纠错模型柱搜索解码得到的语法纠错结果。	26
表 3.2	VERNet 模型采用的数据信息统计。	34
表 3.3	词语级别的语法错误标注标签生成示例。	35
表 3.4	基于 BERT 和长短期记忆循环神经网络 (LSTM) 的语法错误检查模型在 FCE 数据集 ^[65] 上的效果。	39
表 3.5	基于 BERT 的不同语法错误检查模型的效果。	40
表 3.6	不同语法错误检查模型在词级别语法纠错质量评估任务的结果。	40
表 3.7	不同语法纠错质量评估模型在句子级别质量评估效果。	42
表 3.8	语法纠错质量评估模型打分与真实语法纠错评价指标得分 ($F_{0.5}$) 之间的皮尔逊相关系数。	43
表 3.9	VERNet 模型对语法纠错系统的影响。	44
表 3.10	融合多种基于 BERT 的语法纠错质量评估模型对语法纠错效果影响。	45
表 3.11	VERNet [†] 模型提供的语法纠错结果样例分析。	48
表 4.1	事实验证数据集 FEVER 的训练集、验证集和测试集划分。	61
表 4.2	事实验证准确性。	64
表 4.3	证据句子检索准确性评测结果。检索证据的效果是通过官方评价指标 Precision , Recall 和 F1 来进行评测 ^[103] 。	65
表 4.4	不同模型在多证据推理场景和单证据推理场景的事实验证效果。	66
表 4.5	多证据联合推理样例分析。	69

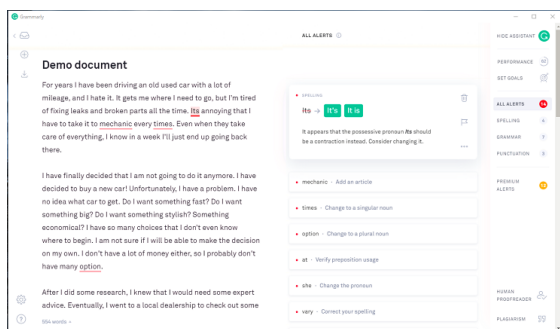
表 5.1	通用领域事实验证数据集 FEVER 以及面向科学领域数据集 SCIFACT 的训练集、验证集和测试集的划分。	79
表 5.2	SciKGAT 在面向专业领域的事实验证任务上的效果。	81
表 5.3	在 SCIFACT 验证集上文档检索效果。	82
表 5.4	TREC COVID ^[143] 文档检索比赛中第二轮无人工干预组各机构排名。 ..	82
表 5.5	在面向新型冠状病毒信息检索任务 TREC COVID 中检索模型的消融实验。	83
表 5.6	在 SCIFACT 验证集上句子检索效果。	83
表 5.7	不同模型在 SCIFACT 开发集上的事实验证效果。	84
表 5.8	事实验证结果样例分析。	85

第1章 引言

1.1 研究背景和意义

校对是文章、图书等刊物出版工作中的重要程序，校对工作需要按照出版规则和要求，发现并且能够及时更正排版或编辑时候所出现的问题。因此，对文章或图书等刊物进行校对是一项极为细致且重要的工作，并成为了这些刊物在出版过程中不可或缺的一个环节。对于文本校对的定义最早可追溯到西汉时期，西汉学者刘向对“校讎”（即校对）有如下定义“讎校，一人读书，校其上下，得谬误为校。一人持本，一人读书，若怨家相对”。这句话说的意思就是，对于文章的校对可以通过通读全文发现刊物中自身存在矛盾的地方并改正，另一个则是通过对比不同的刻本找到不同版本的差异以此来对文章进行修改。

在文本校对中，语法错误校对以及事实验证是两个重要任务，其能够保证文本在语法层面以及语义层面的正确性。权威杂志或刊物对一篇文章都会雇佣若干个编辑对其进行校对，从而能够保证出版物的正确性、合理性以及权威性，进而一定程度上避免了错误信息的传播以及对于公众的误导。然而由于文本校对工作往往是一个比较细致的工作，人们在对文本校对时总是过分相信自己的直觉或者潜在的印象，亦或是文本校对人员所处的环境以及心情等等都会导致漏掉部分文本错误，从而降低校对的质量。此外，随着互联网的发展，网络新闻、报纸以及电子杂志多如牛毛，这些电子刊物大多都是由人工手动输入，因此，不可避免地产生一些错误。据相关研究显示，国家新闻出版部门公布的图书检查合格率，2012年为88.4%，而到2014年即下降为82.5%^[1]。因此，仅仅依靠人类编辑对如此大规模的文本进行校对是不现实的，而且会降低校对的质量。



(a) Grammarly 软件。

Once upon a time, there were a quick brownn fox.

He jumped over the lazy dog.

The lazy dogs is sleeping.



(b) WhiteSmoke 软件。

图 1.1 一些商用文本校对工具。

自然语言处理 (Natural Language Processing, NLP) 是人工智能的核心研究领域之一, 在利用计算机技术对自然语言进行处理的过程中发挥了重要的作用, 也衍生出众多的应用, 例如: 机器翻译, 信息检索, 自动问答, 文本自动校对等等。随着深度神经网络以及预训练语言模型的发展, 自然语言处理技术对于文本的理解能力以及语言模型的建模能力进一步被提高。因此, 将现有的更为先进的自然语言处理技术应用到文本校对中, 以此来实现更加有效的文本自动校对算法, 对于出版单位、政府、和个人都有着重要的价值以及意义。

利用自然语言处理技术自动化地对文本进行校对能够帮助人们减轻工作量, 同时能够为文本的质量保驾护航。相对于人工校对来讲, 计算机自动校对效率高、速度快并且效果稳定。同时, 针对于自动文本校对任务, 其衍生出一系列的商用软件, 如图1.1所示。文本自动校对可以在不同的文章出版场景发挥其作用, 例如, 可以实现文本自动校对系统以此来检测语法错误^[2], 对一些与政治新闻有关的文本进行事实方面的校对^[3-4], 或者对于科学论文中的句子进行自动评分, 以此来提示用户需要修正的地方^[5]。

1.2 英文文本校对任务

英文文本校对任务包含了很多部分, 例如: 语法错误修正、拼写纠错、句子简化、事实验证、句子压缩以及句子转述等等^[6]。而在英文文本校对的诸多任务中, 语法错误校对以及事实验证两个任务作为两个最重要的任务能够保证文本在语法层面以及语义层面的正确性, 因此受到了学术界的广泛关注。语法错误校对任务主要希望能够对行文进行语法层面的修改, 从而保证文本符合语法规则并保证行文的流畅性。而对于事实验证来讲, 更多的是对文本的语义信息进行验证, 判断其是否满足既有的事实知识。尤其是针对一些特定写作背景的文章, 例如新闻等, 事实验证十分重要^[4]。在事实错误校对的过程中尤其需要注意在人名、职名、地点等重要方面进行校对, 以免出现事实差错以及错误信息的传播^[7]。下面我们就语法错误校对以及事实验证这两个任务进行详细介绍。

1.2.1 语法错误校对

语法错误校对主要是通过对文章进行语法层面的错误识别或修改。现有工作对语法错误的类型进行了分类, 其中包含了约 28 种语法错误类型^[8]。为了能够对不同种类的语法错误进行识别亦或是修改, 如图1.2所示, 自然语言处理的相关工作提出了两种任务来实现英文语法的校对, 其中一个为语法错误检查 (Grammatical Error Detection, GED)^[9-10] 任务, 其旨在识别语法错误从而提示用户可能需要修改

的词语；另一个任务则为语法纠错任务（Grammatical Error Correction, GEC）^[8,11-13]，该任务进一步希望能够对一句有语法错误的句子进行修改。

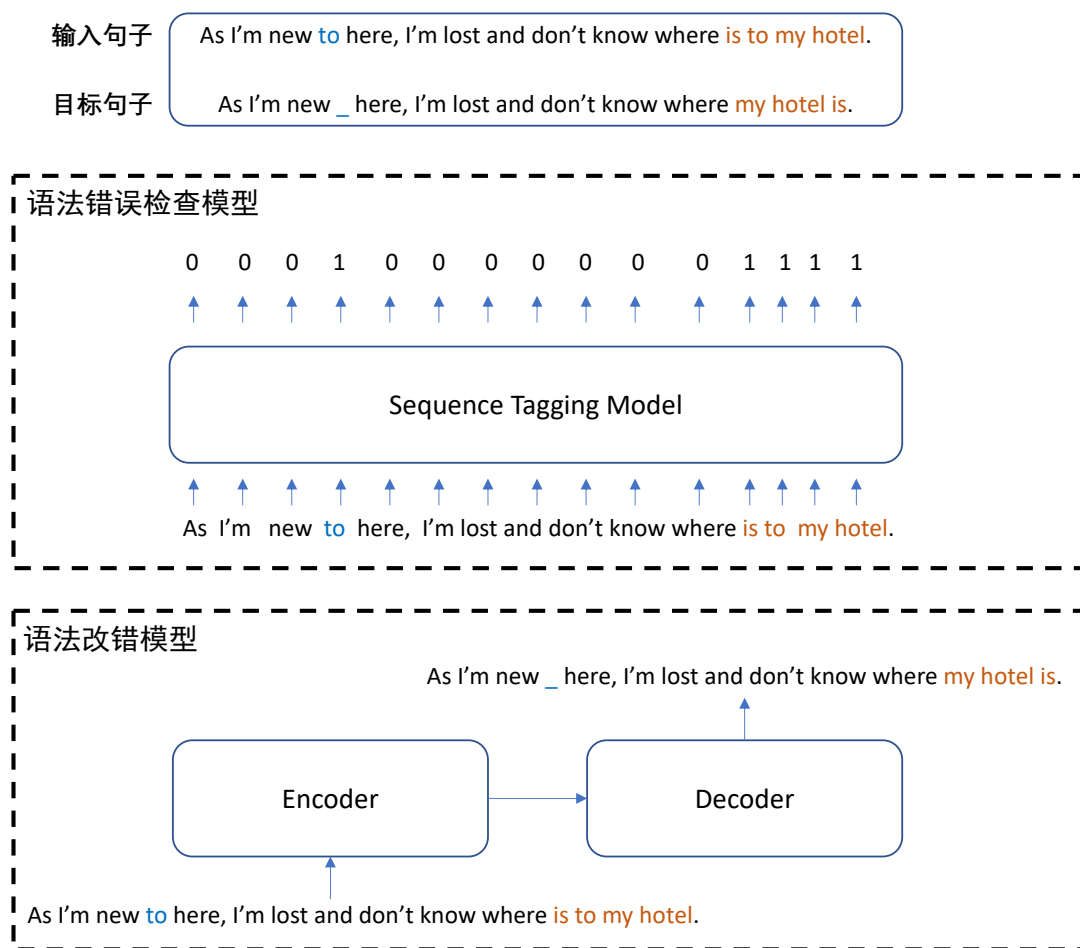


图 1.2 语法错误检查任务以及语法纠错任务示意图。

我们首先介绍语法错误检查任务。针对于给定的包含语法错误的句子，语法错误检查模型希望能够将句子中包含语法错误的词语进行标注，并将其分类为正确和错误，从而能够对用户起到提示作用。现有的语法错误检查模型往往采用长短期记忆循环神经网络（Long Short Term Memory, LSTM）^[14] 来对句子进行编码获得词语的上下文表示，进一步将其抽象成一个序列标注任务，从而对词语进行语法正确性标签标注。受益于预训练语言模型的发展，诸如 BERT^[15] 之类的预训练语言模型也在语法错误检查任务上展现了其良好的效果^[16]。

另外一种语法校对任务则是语法纠错任务，该任务的目的是在不改变句子原意的情况下，将句子的语法错误进行修正，使得句子更通顺、流畅。不同于语法错误检查任务，在语法纠错任务中我们则希望模型能够给出语法纠错结果。因此，现有工作大多应用了基于序列到序列的生成模型来完成对给定句子的改写，从而能

够针对输入句子中不同类型的语法错误进行修正。受益于诸如 Transformer^[17] 之类的深度神经网络模型的发展，基于生成的语法纠错模型得到了充分的研究，并希望能够通过语料增广以及预训练技术来强化深度神经网络模型对自然语言的建模能力，进而提升神经网络模型在语法纠错任务上的效果。受到了一些预训练语言模型的启发^[15,18]，当前针对语法纠错的语料增广技术通常通过在正确的句子中引入噪音的方式构建伪语料并训练语法纠错模型，进而增强语法纠错模型对于错误句子的识别以及修正能力。但是，由于语法纠错语料增广任务相比较其他任务，如自动问答任务以及信息检索任务，更加简单。因此，现有的语法纠错预训练方法相比较通用预训练语言模型能够很好地针对语法纠错任务本身进行数据增广并使得模型在语法纠错任务上达到更好的效果。但是这部分增广语料能力有限，在增广语料达到一定规模后，即使使用更大规模的增广语料预训练语法纠错模型，其效果也没有更显著的提升^[19]。

进一步，已有的工作考虑到语法纠错任务的特点，即输入句子和输出句子大多数词语是不变的，以此来进一步提升语法纠错模型的效果。例如：利用语法错误检查模型识别出有语法错误的部分后利用生成模型有针对性地生成正确的部分^[20]，对语法纠错模型得到的语法纠错结果进行质量评估进而提升语法纠错效果^[21]，将语法错误标注信息加入到语法纠错模型的训练过程中^[22] 等等。本文从其中一方面出发，考虑通过语法纠错质量评估来提升语法纠错效果，并且侧重于如何能够实现更好的语法错误检查模型来提升语法纠错质量评估的效果。

1.2.2 事实错误校对

《南方生活》(Southern Living)的主管阿什利·亚瑟(Ashley Arthur)曾说过信誉对杂志的成功至关重要，无论文章或者杂志的主题是什么：“对我而言，最重要的是事实。如果读者不信任我们，他们将不会购买该杂志”。对于出版的文章以及出版社来讲，保证事实层面的正确性也是极为重要的，因此经常会进行事实验证(Fact Verification)来保证文章的正确性以及真实性。

随着网络传媒的快速发展，传播文章或者消息的途径越来越多，文章中的各类事实错误也如影随形，其中所存在的事实错误往往会造成重大影响。针对新闻等文章，由于竞争激烈会导致调查不够深入亦或是写作仓促从而在政治方面出现严重的错误，造成严重后果^[4,23]。例如，2010年12月30日出版的《人民日报》第4版的文章标题将国务院总理温家宝的姓名错印成“温家室”，最终被校对出来避免了不良的社会影响^[3-4,23]；一些杂志中也存在着事实性错误，《名利场》(Vanity Fair)杂志中将女演员朱莉娅·罗伯茨(Julia Roberts)已故父亲的名字误写为沃尔特·莫特斯(Walter Motes)而不是其真实名字沃尔特·罗伯茨(Walter Roberts)；网

络上还有一些人为了博人眼球会故意伪造一些违背科学事实的文章导致了错误信息的广泛传播，并造成了经济损失，如“5G 信号塔会传播新型冠状病毒肺炎”等。因此，针对于文章的语义层面的校对也是至关重要的，其确保了文章的真实性以及可信性，避免了错误信息传播所造成的影响。

在文章出版过程中，事实错误校对委员会仔细检查文章中的每个句子。为了审阅相关的文章或者对其中的数据进行校对，他们经常会访问相关网页，追溯相关的参考文献，甚至将文章递交给独立专家来进行审查。事实校对工作需要细致地校对，然而人工标注员总是过分相信自己脑海中的潜在印象，导致部分事实错误不能够被识别。此外，面对互联网上的海量文本，雇佣人工进行事实错误校对是难以实现的。因此，需要有一个自动化事实验证工具来辅助人们进行事实验证。

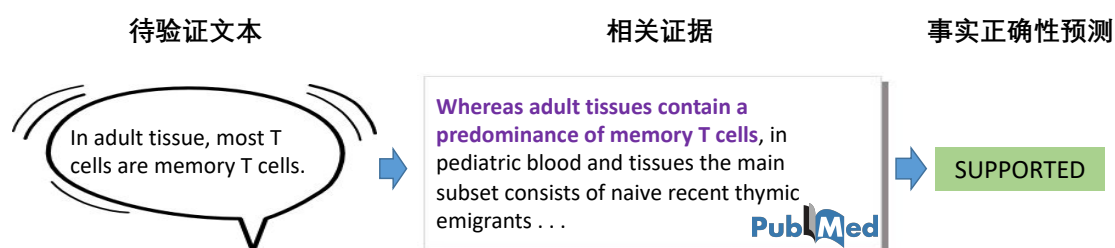


图 1.3 事实验证流程示意图。

一些工作将该任务定义为事实抽取和事实验证 (Fact Extraction and Verification)^[24]。如图1.3所示，我们展示了一个事实验证流程的示意图。在事实验证任务中，我们首先需要通过在一些可信语料库，比如维基百科以及 PubMed 等，可信文档集合中查找与待验证文本有关的事实验证证据，从而对待验证文本进行事实验证。每一个待验证的句子会被划分为支持，拒绝以及没有足够证据验证三类。

具体来说，类似于开放域问答的相关工作^[25]，现有的事实验证工作^[24,26-28]经常使用一个三步流水线模型来实现事实验证系统，其中包括了：文档检索、句子检索以及事实验证。其中，对于文档检索，现有的工作一部分通过实体链接技术识别出与待验证文本相关的维基百科页面^[26-27,29]，另外一部分则是通过信息检索的相关技术，采用例如 BM25 等模型实现文档级别的检索^[30-31]。句子检索任务则是通过对检索到的文章的句子利用信息检索以及短文本匹配技术对相关句子进行筛选，从而获得事实校对证据。现有的事实验证方面的工作往往侧重于事实验证这一步骤上。

受益于自然语言推理方向 (Natural Language Inference, NLI) 的发展，现有的事实验证模型采用了自然语言推理模型，并且根据检索到的证据对当前文本进行

事实验证标签分类^[24]。不同于自然语言推理任务，事实验证任务更加侧重如何对多条事实验证证据进行建模以达到对待验证文本进行联合事实验证的目的^[26-27]。随着预训练语言模型的发展，其在自然语言推理方向展现了良好的效果，能够更好地理解文本的语义信息，因此，当前的事实验证模型往往采用预训练语言模型实现对于文本的编码，进一步提升了事实验证的效果^[28]。

1.3 融合富信息的英文文本自动校对

为了实现对英文文本的自动校对，我们针对语法错误检查任务以及事实验证任务设计了统一的框架，如图1.4所示。我们期望通过融合不同方面的知识来实现融合富信息的英文文本自动校对。



图 1.4 融合富信息的英文文本自动校对方法。

正如我们前面所介绍的，英文文本自动校对任务分为语法错误校对以及事实错误校对两种。在人类进行文本校对的过程中，其往往都会参考一些证据来判断当前阅读的句子是正确的还是错误的。例如，词典中给出的例句，知识库中给出的相关的事实证据以及一些专业领域中的参考论文等等。这样多样化的富信息能够辅助模型对文本进行语法以及语义层面的校对，从而保证文本的语法和事实方面的正确性。对于事实验证任务，我们则需要找到世界知识库或者领域知识库中与当前待验证文本相关的证据，进一步利用模型验证当前的输入文本的事实正确性。

随着诸如 BERT^[15] 之类的预训练语言模型的迅速发展，深层神经网络模型能够更好地建模语言知识并且理解文本的语义信息，进一步保证了英文文本自动校对的有效性。然而，如图1.5所示，现有的文本校对工作在 **信息层**（文本校对证据获取）、**算法层**（融合多文本校对证据辅助文本错误检查）以及 **应用层**（如何面向专业领域进行文本校对）三个层面还存在很多问题。

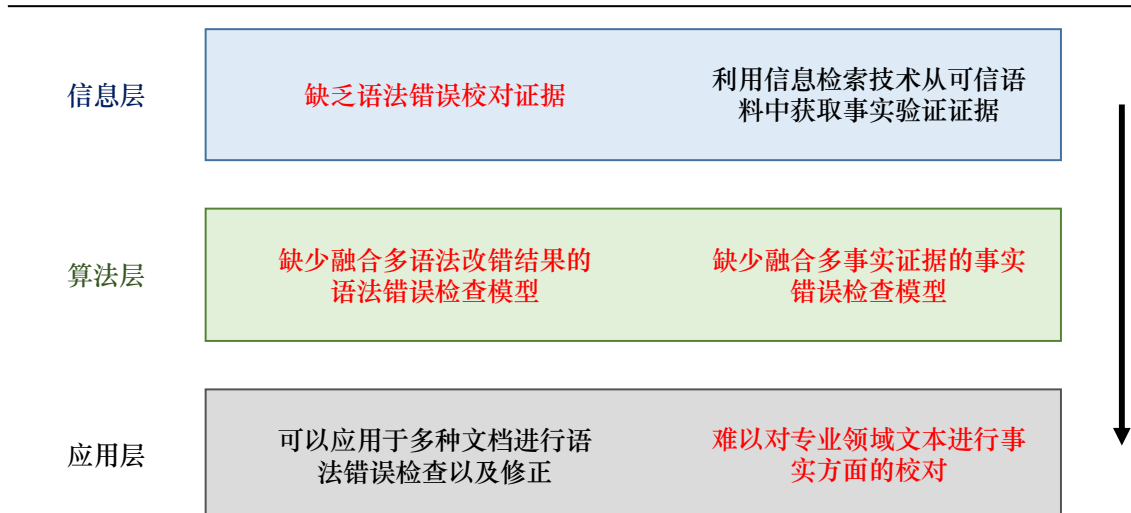


图 1.5 文本自动校对所面临的挑战。

首先，在信息层，为了实现文本自动校对，我们需要找到能够辅助模型验证当前文本语法以及事实错误的证据。前人工作对事实验证的校对证据获取有了很好的定义^[24]，并采用了一些信息检索的工作^[32]来实现针对于可信语料库，诸如维基百科，的事实验证证据获取^[26,28]。然而，如何针对于语法错误校对任务获取语法校对证据还未有人进行充分探究。因此，**缺乏语法错误校对证据**是在文本自动校对任务的信息层中的主要挑战，我们在第2章中给出了详细的解决方案。

其次，在算法层，我们需要通过融合信息层所提供的文本富信息来辅助文本进行语法校对以及事实验证。针对于语法校对和事实校对任务，其分别侧重于对语法和语义错误进行检查，因此，分别面临着**缺少融合多语法改错结果的语法错误检查模型**以及**缺少融合多事实证据的事实错误检查模型**两个挑战。我们分别根据语法错误检查任务以及语义级别的事实错误检查任务分别在第3章以及第4章给出我们的解决方案。

最后，在应用层，文本校对工作往往会面向一些专业领域，如医学领域、法律领域以及金融领域等等。在文本校对的两个子任务中，事实验证任务是从语义层面上对文本错误进行识别，因此其更加考验模型对于文本的理解能力。但是由于在专业领域中，相应的标注数据标注成本昂贵，导致标注标签较少。然而，小样本标签不能很好地训练语言模型，限制了语言模型对于专业领域文本的理解能力，进而导致**难以对跨领域文本进行事实方面的校对**。

针对信息层、算法层以及应用层这三个文本自动校对层面，具体来说主要存在以下四个方面的问题：

- **缺乏语法错误校对证据**。参考人类进行语法校对时候的行为，其往往需要借助于词典或者相关例句等包含语言知识的线索辅助模型对当前文本进行语法错误检查。随着语言模型预训练方向的发展，神经网络生成模型在大多数

自然语言处理任务上都显示了良好的效果，尤其是在语法纠错任务上。因此，怎样能够对语法纠错模型进行充分训练，使其为语法错误校对模型提供一些高质量的语法纠错线索，进而辅助语言模型对语法错误进行检查是一个重要的问题。

- **缺少融合多语法纠错结果的语法错误检查模型。**为了充分利用语法纠错模型所提供的语法纠错线索，我们需要利用多条语法纠错结果辅助语法错误检查。由于每条语法纠错结果都会对输入文本中的每一个词语提供相应的语法纠错方案，且语法纠错方案可能相同或不同。因此，怎样能够从每一个语法纠错结果中抽取并汇总能够支持或反驳当前文本中词语的语法正确性的语法纠错线索，并且在语法层面对当前文本进行语法错误检查是一个重要的研究挑战。
- **缺少融合多事实证据的事实错误检查模型。**在事实验证证据的获取阶段，其往往利用信息检索相关算法在世界知识库中搜索与待验证文本相关的事实证据，因此不可避免地会引入额外的噪音。此外，不同于融合多语法纠错结果的语法错误检查，文本中存在的事实错误往往是一个十分细微的错误，可能是一个年份、一个人名亦或是地名等。由于待验证文本与检索到的事实证据之间过度相似，降低了事实验证模型对于事实错误的识别能力。因此，融合多事实证据的事实验证任务更期望模型能够对待验证文本和证据之间进行细粒度推理以实现更加准确的事实验证结果。
- **难以对专业领域文本进行事实方面的校对。**在事实验证方面，很多需求都是面向于专业领域，比如：医学领域、法律领域等等。然而，由于面向专业领域的标注工作往往需要有很强专业背景的人员参与，因此导致了此部分数据往往十分珍贵且稀少^[31]。小规模的数据限制了面向专业领域的事实验证模型的效果。然而，当前事实验证的能力往往来源于其编码器，即预训练语言模型的能力^[26,28,31]。因此，如何能够进一步提升我们的预训练语言模型在这些专业领域上对文本的理解能力，从而提升面向专业领域事实验证模型的效果，也是一个重要的研究任务。

1.4 本文的主要研究内容

如图1.6所示，本文针对融合富信息的英文文本自动校对中的语法错误校对以及事实验证两个任务进行研究，系统性地进行了以下四个工作来解决在融合富信息的英文文本自动校对任务中存在的挑战：

- **语法纠错算法的框架和语法纠错证据获取。**在英文语法纠错任务中，本工作

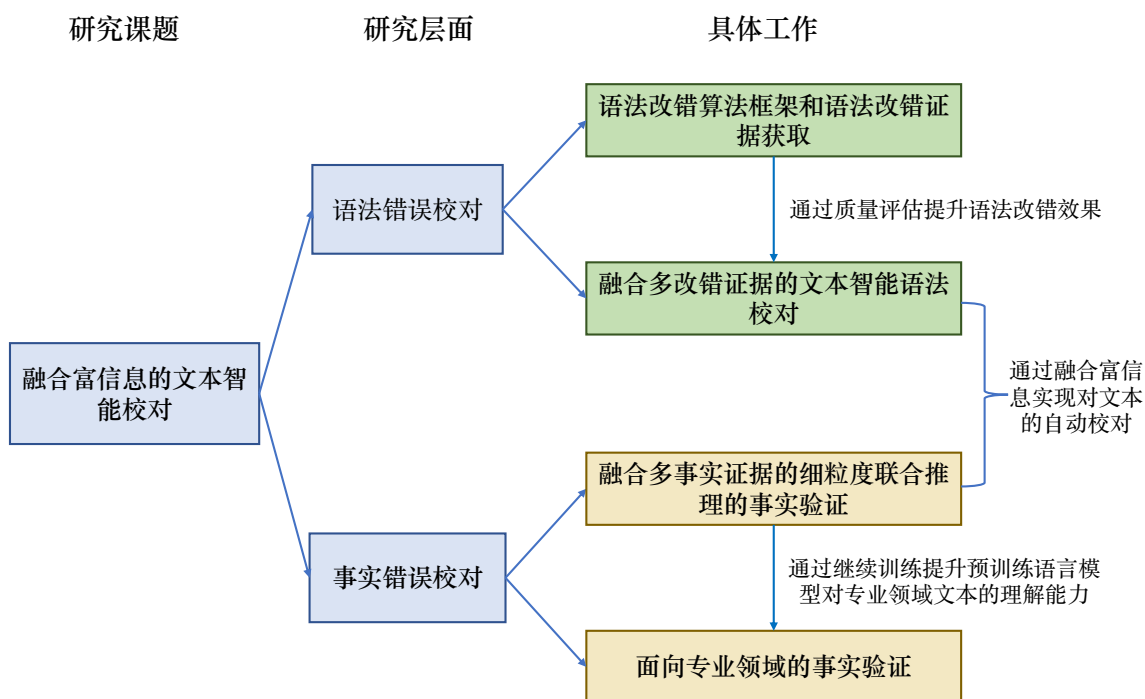


图 1.6 本文研究内容框架。

对现有的通用预训练语言模型以及面向语法改错任务的预训练语言模型进行比较，并确定了针对于英文语法改错较好的预训练方式，从而在语法改错任务中实现了当前最好的基于生成的语法改错模型。进一步我们设计了针对于含有噪音的来自语言学习者论坛上的数据进行筛选的强化学习模型，以此提升了英文语法改错模型的性能。

- **融合多语法改错证据的英文语法校对。**通过上一个工作得到的语法改错模型，我们利用柱搜索解码可以进一步得到若干个语法改错结果，以此为语法错误检查模型提供高质量的语法改错证据，从而能够对潜在的语法错误起到提示作用。因此，在本工作中，我们充分利用了经过良好训练的语法改错模型所提供的语法改错结果，并采用不同的注意力机制从多个语法改错结果中找到有益于验证输入句子和语法改错结果中词语的语法正确性和语法改错正确性的语法改错线索。该工作证明了其在语法错误检查和语法改错质量评估两个任务上的有效性，进一步提升了语法改错的性能。
- **融合多事实证据的细粒度联合推理的事实验证。**在事实验证场景下，由于检索到的证据中包含很多噪音，其中只有部分词语的语义信息可以用来验证文本的事实层面的正确性。因此，本工作提出了一种基于核函数的注意力机制从每一个证据中获取细粒度的语义信息进行多证据的联合推理，并在通用事实验证数据集上达到了良好的效果。

- **面向专业领域的事实验证。**由于对于一些专业领域，诸如：医学领域、法律领域等等，往往更加需要对文本进行事实层面的检查。因此，本工作探讨了对于专业领域的事实验证工作，并提出了两种语言模型训练方法。通过对预训练语言模型进行继续训练进一步提升语言模型在专业领域中对文本的理解能力。该工作在一个面向科学领域的事实验证数据集上显示了其有效性。最后，我们对本文工作进行总结，并对未来研究方向做出展望。

第2章 语法纠错算法的框架和语法纠错证据获取

语法纠错 (Grammatical Error Correction, GEC) 是文本校对的重要工作之一, 其目的是采用自然语言处理技术自动化地对文章中的语法错误进行识别以及修改。近些年, 随着深度学习技术的不断发展, 语法纠错的性能有了很大的提升。当前大多数语法纠错系统都基于神经机器翻译 (Neural Machine Translation, NMT) 的方法, 并应用序列到序列模型 (Sequence-to-Sequence) 生成修改后的句子。不同于机器翻译任务, 很多语法纠错的工作都将语法纠错任务看做一个低资源的生成任务, 并将其关注点放在了如何更好地进行语料增广上面。同时, 随着预训练语言模型不断发展, 其在许多自然语言处理任务中都显示了良好的性能。因此, 在本章中, 我们将大多数语法纠错语料增广技术以及通用预训练语言模型的效果进行比较, 以确定针对于语法纠错任务最好的预训练方式。同时本章工作在前人工作基础上确定了一组更优的训练参数, 进一步在现有的公开语法纠错数据集上进行微调, 达到当前基于序列到序列模型最好的效果。同时我们通过本章所设计的数据筛选策略对包含噪音的语法纠错语料进行筛选, 并将其加入到训练过程中, 进一步提升了语法纠错模型的性能。

2.1 问题描述

语法纠错^[2] 任务是为文本校对所设计的, 该任务希望能够自动化地识别句子中的语法错误并进行合适的修改。

在早期的语法纠错模型中, 大多数语法纠错系统都基于手工规则^[33-34]。然而, 设计相应的语法纠错的规则并解决语法纠错规则之间的冲突是十分复杂的, 并且, 其构建过程中需要大量的人工劳动。虽然, 在语法纠错任务中的一些工作, 例如拼写纠错 (Spelling Check), 依旧使用手工规则来进行语法错误修正。但是, 一些大规模的人工标注语料库的构建^[35-37] 为实现完全数据驱动的语法纠错模型提供了坚实的基础, 并获得了良好的效果。因此, 我们主要研究基于数据驱动的语法纠错模型。

基于数据驱动的语法纠错模型发展十分迅速, 早期的数据驱动的语法纠错模型实现了对部分语法错误的修正工作。在这种方法中, 研究人员通过考虑上下文的语言特征, 对分类器进行训练, 使其能够纠正介词错误之类简单的语法错误。为了能够在多种语法错误上进行修正, 基于统计机器翻译 (Statistic Machine Translation, SMT) 的语法纠错系统^[38-41] 受到了学术界的关注。基于统计机器翻译的语法纠错

方法将错误句子看作是源语言，正确的句子看作是目标语言，通过标注语料中的错误、正确句子对来训练统计机器翻译模型，进而期望基于统计机器翻译的语法纠错模型能够修正多种语法错误。

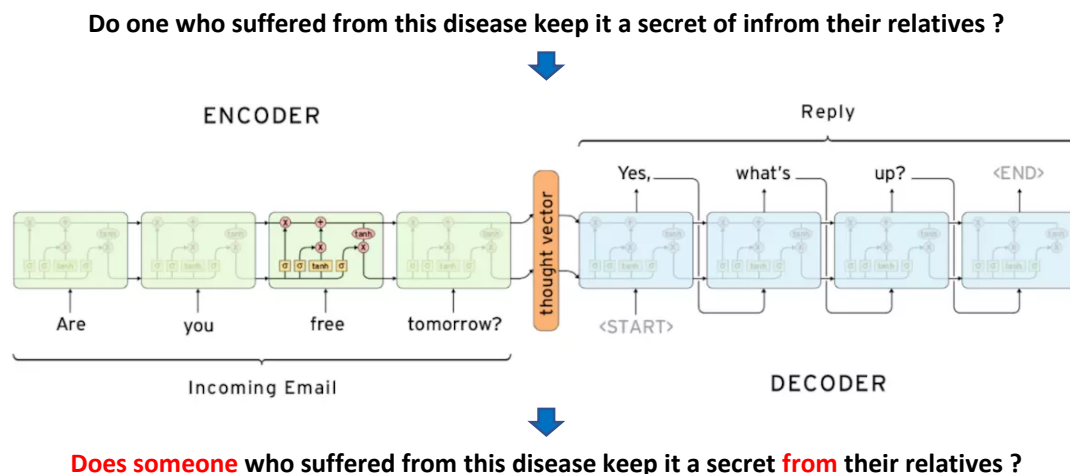


图 2.1 语法纠错任务示意图。

随着深度学习技术的不断发展,如图2.1所示,现有的语法纠错工作经常将语法纠错任务视为一个自然语言生成 (Natural Language Generation, NLG) 任务,同时应用序列到序列的神经网络结构 (Sequence-to-Sequence)^[42] 并通过柱搜索解码生成语法纠错结果。早期的神经网络生成模型通常在编码器和解码器部分采用循环神经网络 (Recurrent Neural Network, RNN) 以及卷积神经网络 (Convolutional Neural Network, CNN) 进行编码和解码,进一步实现一个序列到序列生成模型^[43-44]。许多语法纠错相关研究也应用了这些序列到序列的神经网络结构来实现语法纠错模型^[45-46]。随着神经网络结构的发展,基于 Transformer 的序列到序列生成模型^[17]在许多自然语言生成任务中都显示了其有效性,同时也被用于实现当前最好的基于自回归生成的语法纠错模型^[19,47]。

对于深度神经网络技术来说,数据规模往往决定了模型的效果。然而,现有的语法纠错数据的规模较小,难以充分地训练基于深度神经网络的生成模型。因此,现有的语法纠错工作经常将注意力放在如何更好地进行语料增广上,以此来缓解语法纠错模型的数据不足问题。为了生成大量的训练语句对,已有工作利用不同数据扩充方法根据一个正确句子来生成一个包含语法错误的句子,以此构建训练语料库。其中一些工作对正确文本进行词级别的随机替换、删除以及添加等操作,从而形成一个含有语法错误的句子;而其他一些工作则对无错误的文本应用反向翻译,以将其翻译为不符合语法规则的句子。两种方法都普遍应用于当今的语法纠错系统中,以此来预训练语法纠错模型。此外,语法纠错任务还存在如下一些问

题。首先，随着预训练语言模型的发展，其在自然语言处理任务上也展示了良好的效果，然而，此类预训练语言模型在语法纠错任务上的效果还未被充分的探究。其次，当前的语法纠错的预训练数据很多都是不可获得，又或是仅提供部分模型和测试代码^[19]，因此，需要进一步对模型训练过程进行调参，从而复现出相关的研究结果。在本章中，我们主要有如下三个贡献：

(1) 我们首先探究了几种通用预训练语言模型以及不同的语法纠错语料增广技术在语法纠错任务上的效果，从而确定了最优的语法纠错预训练方式。

(2) 基于当前较好的语法纠错语料增广技术，我们进一步在标注语料库上对预训练语法纠错模型进行微调，并确定了一组更为合适的参数，从而实现了当前最好的基于自回归生成的语法纠错模型。

(3) 在很多模型的训练过程中都没有用到英文学习者网站 Lang8 的语法纠错数据，该数据数量规模较大，但是由于该数据集中的标注都是众包产生的，因此存在很多噪音数据。我们通过采用一些已有方法对数据进行筛选，并将筛选后的数据加入到模型微调过程中，从而提升语法纠错模型效果。

2.2 相关工作

根据前人的工作^[8]，常见的语法错误大约可以分为 28 种，如表2.1所示，我们展示了一些语法错误的例子。当前语法纠错工作大多基于 Transformer 的序列到序列生成模型^[17]来实现当前最好的基于自回归生成的语法纠错模型^[19,47]。

由于在语法纠错任务中，输入句子中绝大部分都是不需要进行修改的，因此，一些工作也基于这个特征将拷贝机制^[48]引入到语法纠错模型中，以便更好地将输入句子和纠错结果进行对齐，从而提升语法纠错的性能^[22]。为了进一步加快生成过程，一些工作还提出了非自回归语法纠错模型。在这里，不同于在机器翻译领域使用的非自回归生成模型^[49]，非自回归语法纠错模型通常将预训练语言模型^[15,50-51]作为编码器对语法错误并行检测和纠正^[52-54]。

已有的语法纠错模型往往将语法纠错任务看做一个低资源翻译任务，同时提出了不同的方法进行语料库增广，从而提升语法纠错模型的效果^[19,55]。其中一些方法利用了不同的语料库，如维基百科的修改历史^[56]、Github 的修改历史^[57]以及混淆词词典^[47]等等。此外，一些工作通过生成模型或回环翻译引入额外的语法错误^[58-60]。Kiyono 等人^[19]进一步融合随机替换以及反向翻译等多种语法纠错数据增广策略，以此提升语法纠错模型预训练效果。

此外，随着预训练语言模型的发展，诸如 BERT^[15]之类的预训练语言模型采用了基于掩码语言模型 (Mask Language Model, MLM) 的预训练策略，进一步提

表 2.1 常见的一些语法错误示例。

描述	例子
冠词或限定语错误	It is obvious to see that [internet → the internet] saves people time and also connects people globally.
错误搭配	Early examination is [healthy → advisable] and will cast away unwanted doubts.
冗余	It is up to the [patient's own choice → patient] to disclose information.
名词单复数错误	A carrier may consider not having any [child → children] after getting married.
动词时态错误	Medical technology during that time [is → was] not advanced enough to cure him.
拼写, 标点等错误	This knowledge [maybe relavant → may be relevant] to them.
指代错误	It is everyone's duty to ensure that [he or she → they] undergo regular health checks.
词形错误	The sense of [guilty → guilt] can be more than expected.

升了神经网络模型在自然语言处理任务上的效果。因此, Kaneko 等人^[61]延续了 Kiyono 等人提出的语法纠错模型^[19], 使用 BERT 对输入句子进行编码, 并将 BERT 编码后的词语向量表示融合到语法纠错模型中, 从而提升语法纠错模型的效果。然而, 一些最新的面向自然语言生成任务的预训练语言模型, 例如: GPT-2^[62] 以及 T5^[63], 其模型效果还没有在语法纠错任务上进行评测。

2.3 语法纠错数据集

我们首先介绍几个最广泛使用的语法纠错数据集, 其中包括了 NUCLE^[36], Lang8^[64], FCE^[65], JFLEG^[66], BEA19^[37] 以及 CoNLL-2014^[8]。各个数据的详细信息如表2.2所示, 这些数据集经常用于训练和评测语法纠错模型。接下来我们对每个数据集进行简单的介绍。

NUCLE 数据集。英语学习者语料库 (NUCLE) 是由新加坡国立大学所构建的一个可免费使用, 并以研究为目的开放语法纠错数据集。NUCLE 数据集包含 1,414 篇由在新加坡国立大学就读的亚洲本科生撰写的论文, 这些文章包含了 1,220,257 个词, 其中 46,597 个带有语法错误。由于该数据集都是来源于亚洲学生的论文, 因此导致了写作的主题, 写作熟练度和数据来源者的母语的多样性较低。

表 2.2 语法纠错数据集信息统计。

数据集	句对数量			语法纠错目标	标注数量	评价指标
	训练集	验证集	测试集			
Lang8 ^[64]	1,371,668	-	-	流利度修改	1-8	-
FCE ^[65]	28,350	2,191	2,695	最小修改	1	F _{0.5}
BEA19 ^[37]	34,308	4,384	4,477	最小修改	1	F _{0.5}
NUCLE ^[36]	57,151	-	-	最小修改	2	-
CoNLL-2014 ^[8]	-	-	1,312	最小修改	2	F _{0.5}
JFLEG ^[66]	-	754	747	流利度修改	4	GLEU

FCE 数据集。 FCE 数据集 (First Certificate in English Corpus) 是剑桥学习者语料库 (CLC) 中的一部分^[67], 它包含了 1,244 篇英语学习者为回答 FCE 考试问题而写的文章。FCE 数据集的来源者的母语更具有多样性, 句子熟练度和主题更加丰富。

Lang8 数据集。 Lang8 数据集是从语言学习者社交网站^①收集到的英语部分的语法纠错数据集。其中, 语言学习者发布需要进行语法错误修正的文章, 而相应母语使用者则对给定的文章进行修改。尽管 Lang8 数据集包含很多真实世界中包含语法错误的句子, 然而由于对文章进行修改的用户语言能力不同, 以及标注标准不同, 导致了数据标注质量的降低。

BEA19 数据集。 BEA-2019 语法纠错共享任务是最近被提出来的, 该任务引入了 Write & Improve 语料库 (W & I) 和 LOCNESS 语料库以此来构建 BEA-2019 语法纠错共享任务的相关数据集。W & I 由来自 Write & Improve 的 3600 篇带标注的被提交的论文组成, Write & Improve 是一个在线网络平台, 可帮助母语非英语学习者写作。该部分包括了 3600 篇文章, 其中分布在训练集, 开发集和测试集中的文章数量, 分别为 3000、300 和 300 篇。对于另外一个语料库 LOCNESS, 其中包含了由英、美两国大学生撰写的约 400 篇文章, 因此该部分数据所包含的语法错误都是来自于英语母语学习者。在构建 BEA-2019 评测任务数据集时, LOCNESS 数据仅仅被包含在了开发集和测试集中。

CoNLL-2014 数据集。 CoNLL-2014 共享任务是由新加坡国立大学发布的专门用于语法纠错评测的评测任务。这个评测任务给定了其任务定义, 数据集定义, 并描述了共享任务中使用的评估指标和评分细节。该评测任务首先提出要将主要的评价指标定为 F_{0.5} 以更加强调语法纠错结果的准确性。该数据包含了来自两名母语为英语的人类标注者给出的标注结果作为语法纠错结果的评测。该部分数据的

① <https://lang-8.com>

构建过程中招募了 25 名母语非英语者，他们撰写了新文章，用作构建 CoNLL-2014 共享任务中的评测数据集。

2.4 语法纠错预训练策略

本小节详细地介绍几种语法纠错的预训练策略。主要分为针对语法纠错任务的预训练方式以及通用语言模型的预训练方式。其中通用的预训练语言模型包含了 GPT-2^[62] 以及 T5^[63]。在这里我们主要介绍一些针对于语法纠错任务本身的预训练语料增广策略：

随机替换^[22]。在基于随机替换语料增广策略训练的语法纠错模型中，其语法纠错模型采用了带有拷贝机制的 Transformer 模型，随后在 One Billion Words Benchmark^[68] 语料库的句子上进行随机替换、修改、删除以及增加词语从而生成包含语法错误的句子，以此进行语法纠错语料增广。此种语料增广方式相对比较简单，因此，在实验中作者将 One Billion Words Benchmark 中的句子以不同的随机种子进行若干次词语随机替换、修改、删除以及增加操作，进而得到若干个包含语法错误的句子。

混淆词典^[47]。在基于混淆词典预训练策略的语法纠错模型中，作者提出了一种简单有效的无监督语法错误生成方法，该方法基于从拼写检查器 Aspell^①中找到的易混淆词语，以此构建出混淆词典来对词语进行替换，从而生成包含语法错误的句子。易混淆词典如表2.3所示。合成数据用于预训练 Transformer 序列到序列模型，该语料增广方案不仅可以提升语法纠错模型的效果，同时为实现实用的语法纠错系统提供了可能。该工作使用了 News Crawl^②语料库生成预训练增广语料，这个语料库主要是由在线报纸中摘录的文章所构建的^[69]。

表 2.3 由拼写检查器 Aspell 所提供的潜在的混淆词示例。

正确词语	易混淆词语
word	wordy, words, world, Ward, Wood
apple	Apple, apply, applier, apples, applet
suggest	smuggest, snuggest, suggests, suggester, saggier
like	Luke, liker, liked, liken, likes
awful	awfully, afoul, lawful, earful, EFL
fully	filly, foully, full, folly, fill

① http://aspell.net/0.50-doc/man-html/8_How.html

② <http://data.statmt.org/news-crawl>

反向翻译^[60]。该工作通过使用单语语料库来合成包含语法错误的句子。该工作通过使用生成模型，将无语法错误的句子作为输入，将有语法错误的句子作为期望的输出，以此来训练生成模型。针对给定的无语法错误的语料库，该工作采用基于柱搜索解码的方式合成人类评估人员几乎无法与非合成语法纠错数据区分的包含语法错误的句子。

维基百科修改历史^[56]。由于维基百科提供了所有页面的修订历史信息，因此该方法提出采用维基百科的编辑记录来自动生成真实的语法纠错数据。维基百科会记录每一个版本的变化，里面包含大量的人工修订信息，因此可以通过维基百科相邻版本之间的变化来对语法纠错数据进行增广。作者从相应版本的维基百科页面中提取并对齐文本，删除非文本元素。然后，该方法以一定概率保留对齐的文本，跳过未对齐的文本，并形成维基百科修改句对，作为语法纠错增广数据。该语法纠错数据中输入序列由较旧版本的维基百科提供，目标序列由较新版本的维基百科所提供。该语料中包含一些语法错误，但绝大多数都不是语法错误。因此，在收集到增广数据之后，在输入序列中引入拼写错误，以及随机进行字符的删除，插入，替换或相邻位置词语换位等操作增加输入序列中包含的噪音。

回环翻译^[56]。该方法通过回环翻译^[70] (round-trip translation) 将翻译模型的弱点和翻译固有的各种歧义作为噪声引入到正确语料中，从而能够为语法纠错任务生成大规模的平行语料。因此，该工作利用回环翻译生成的语料训练语法纠错模型，并将原始句子作为目标序列，而经过回环翻译输出的句子作为相应的输入序列。该工作使用了四种语言，法语，德语，日语和俄语作为桥接语言，进一步将多种广泛使用的语言的翻译模型的噪音引入到正确句子中。与真实语法纠错语料相比，这些基于回环翻译方法生成的句子仅对其中一小部分进行了修改，为了解决这一不足，该工作与基于维基百科修改历史的语料增广方法类似，同样通过插入，删除和相邻位置词语换位等操作在文本中引入噪音。

综合增广策略^[19]。在进行语法纠错语料增广时，如前面所介绍，有多种语料增广的方式可以考虑。因此，该工作从下面几个方面考虑并进一步探究如何更好地对语法纠错模型进行预训练，例如生成伪数据的方法以及训练策略等等。该工作通过探究并组合不同的语法纠错语料增广策略，例如：反向翻译^[60] 策略以及随机替换^[22] 策略，从而确定了一组更为合适的伪语料生成策略。

基于编辑隐状态的生成模型^[71]。大多数数据增强方法主要是通过直接在正确句子中对词语加入噪音来实现的，这导致所产生的语法错误缺乏多样性。因此，该工作提出了一种新的数据增强方法，该方法可以将噪声应用于句子的潜在表示。通过编辑句子的潜在表示，进而利用生成模型产生包含各种类型的语法错误的句子。

该工作同时结合一些预定义的规则，进行相应语法错误类型的数据增广。该方法进一步提高了现有语法纠错模型的性能和鲁棒性。

2.5 基于强化学习的噪音数据筛选方法

正如我们前面所介绍的，虽然来源于英文学习者网站的 Lang8 数据集中句对数量很多，但是由于标注者的语言能力不同，导致一些预训练语法纠错模型在微调时候并未用到该部分数据^[19]。为了实现更好的语法纠错模型，我们借鉴了一些相关的基于强化学习的数据筛选方法^[72]来完成对于 Lang8 数据的筛选工作，如图 2.2 所示。

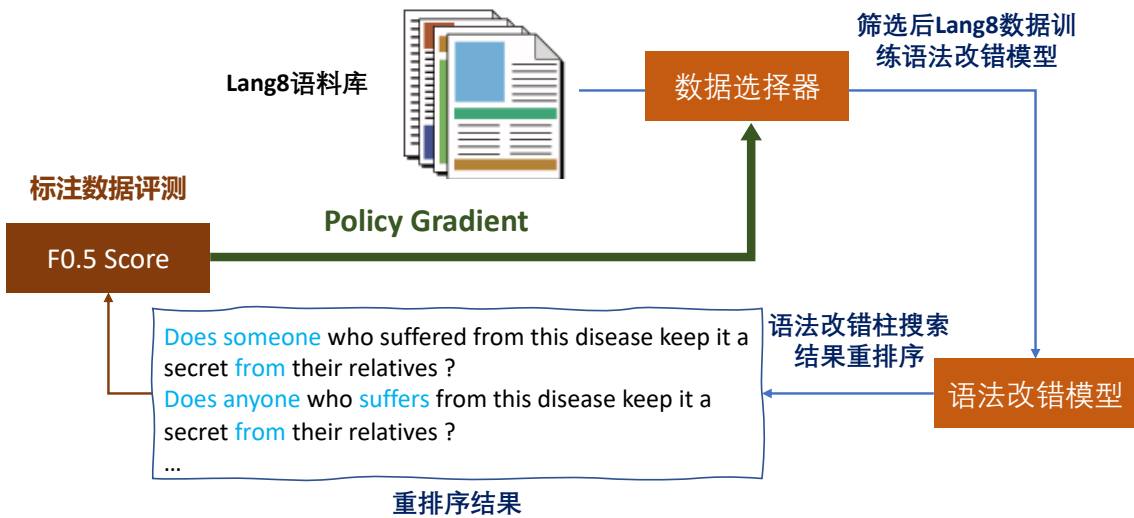


图 2.2 基于强化学习的语法纠错语料筛选方法。

基于 BERT 的数据筛选器。当给定第 t 时刻输入到模型中的批处理数据，其中包含 n 组语法错误、正确句子对。在第 i 组句对 $\langle s_i^t, t_i^t \rangle$ 中， s_i^t 表示待修改句子， t_i^t 表示修改后句子。我们使用预训练语言模型 BERT^[15] 来对句对 $\langle s_i^t, t_i^t \rangle$ 进行编码，并得到其全部词语的隐状态表示 H_i^t ：

$$H_i^t = \text{BERT}([\text{CLS}] \circ s_i^t \circ [\text{SEP}] \circ t_i^t \circ [\text{SEP}]), \quad (2-1)$$

其中 \circ 代表拼接操作，“[CLS]”和 “[SEP]”标记句子的开始与结尾。在这里我们使用 “[CLS]”的隐状态表示 $H_i^t([\text{CLS}])$ 来表示句对 $\langle s_i^t, t_i^t \rangle$ 。随后我们可以算出每个句对被选择的概率：

$$f(s, t) = \text{softmax}_y(\text{Linear}(H_i^t([\text{CLS}]))), \quad (2-2)$$

这里标签 y 表示句对 $\langle s_i^t, t_i^t \rangle$ 被选择 ($y = 1$) 还是被过滤 ($y = 0$)。

数据筛选策略。在这里我们根据数据的选择概率 $f(s_i^t, t_i^t)$ 进行数据筛选，相应

数据筛选动作 a_i^t 被分为 0 和 1 两类，从而设计出如下的数据筛选策略。

$$\pi(s_i^t, t_i^t) = f(s_i^t, t_i^t)^{a_i^t}. \quad (2-3)$$

奖励函数的计算。在奖励函数的计算中，我们使用语法纠错模型（这里我们使用 GPT-2 模型）对基线语法纠错模型^[19]得到的柱搜索结果进行排序，排名第一的语法纠错结果用来计算相应的语法纠错评价指标 $F_{0.5}$ ，从而得到对于当前时刻 t 对应的语法纠错模型的得分 $F_{0.5}^t$ ，以此来将语法纠错评价指标 $F_{0.5}$ 引入到模型的训练中来，并根据 $t-1$ 时刻和 t 时刻的 $F_{0.5}$ 差值作为当前时刻的奖励函数 r^t ：

$$r^t = F_{0.5}^t - F_{0.5}^{t-1}, \quad (2-4)$$

在这里，我们根据 t 时刻模型给出的对批处理数据的数据筛选动作序列 $\{a_1^t, \dots, a_n^t\}$ 对 t 时刻的批处理数据筛选，然后得到的被筛选数据用来训练语法纠错模型，从而对当前语法纠错模型进行更新并得到下一时刻的语法纠错模型参数。

数据选择器参数更新。进一步我们利用 policy gradient^[73] 来更新数据选择器模型的参数 θ^t ：

$$\theta^{t+1} \leftarrow \theta^t + \alpha \sum_{i=1}^n r^t \nabla_{\theta^t} \log \pi_{\theta^t}(s_i^t, t_i^t), \quad (2-5)$$

这里 α 为更新模型参数时候的学习率。

2.6 实验

这一节中，我们将介绍在语法纠错任务中实验所用到的数据集，实验设置以及最终的实验结果。

2.6.1 实验设置

在我们之前提到的相关语法纠错数据增广策略中，综合增广策略^[19] 显示出其良好的效果，但是由于其源代码^①并未给出模型训练命令，我们尝试采用论文中给出的训练参数发现效果并不理想，因此我们通过调试并比较了不同训练参数，最终得到训练效果较好的一组参数如表2.4所示。

在实验中，我们也评测了两种预训练语言模型在语法纠错任务上的效果。我们选择了两种当前最先进的生成模型，GPT-2^[62] 以及 T5^[63]，作为基线模型。在所有实验中，我们使用 GPT-2 的“Medium”版本和 T5 的“Base”版本，并继承了 Huggingface 的 PyTorch 实现^[74]。此外，我们采用 Adam^[75] 用于参数优化，学习率设置为 $5e-5$ ，批处理大小设置为 16，并且在训练过程中将步长累计为 4。此外，我

① <https://github.com/butsugiri/gec-pseudodata>

表 2.4 综合增广策略^[19] 在标注语料微调阶段所采用的参数。

参数	实验设定
Model Architecture	Transformer (“big” 设置)
Optimizer	nag
Attention Dropout	0.2
Clip Norm	2
Dropout	0.2
Loss Function	Label smoothed cross entropy (smoothing value: 0.1)
Learning Rate	5e-05
Learning Rate Period Updates	73328
Learning Rate Scheduler	triangular
Learning Rate Shrink	0.95
Max Epoch	50
Max Learning Rate	0.004
Max Tokens	3000
Relu Dropout	0.2
Beam Search Size	5

们将输入序列和目标序列的最大句子长度设置为 80。

在基于强化学习的数据筛选策略中，我们利用 BERT 作为数据筛选器，学习率设置为 5e-5，批处理大小设置为 16，我们将输入序列和目标序列的句子最大长度设置为 100。

2.6.2 不同预训练方法在语法纠错任务上的效果

在本实验中，我们比较了不同的预训练方法在语法纠错任务上的效果，我们在两个广泛使用的语法纠错数据集 CoNLL-2014 以及 BEA19 上评测模型的效果，如表2.5所示。

通过实验我们发现，相比较其他语法纠错语料增广策略，综合语料增广策略^[19]以及基于编辑隐状态的生成模型^[71] 效果表现最佳。由于 Kiyono 等人^[19] 并未开源训练代码，因此我们重新调试并实现了我们自己的基于综合语料增广策略预训练的语法纠错模型。我们所实现的模型在 BEA19 相关的评测数据中达到最好的效果，同时，在 CoNLL-2014 数据上也达到了与最近发表的基于编辑隐状态的生成模型的数据增广方法^[71] 几乎一致的效果，从而证明了综合语料增广策略^[19] 的有效性。

表 2.5 不同预训练方法在语法纠错任务上的效果（单一模型比较）。

预训练方法	CoNLL-2014			BEA19		
	Precision	Recall	F0.5	Precision	Recall	F0.5
无语料增广	67.06	22.44	47.98	47.1	30.2	42.37
反向翻译 ^[60]	57.2	32.0	49.4	-	-	-
维基百科修改历史 ^[56]	64.5	36.2	55.8	-	-	-
回环翻译 ^[56]	64.4	38.4	56.7	-	-	-
随机替换 ^[22]	67.74	40.62	59.76	-	-	-
混淆词典 ^[47]	-	-	61.3	-	-	64.16
综合策略 ^[19]	68.59	44.87	62.03	66.20	61.48	65.20
基于编辑隐状态的生成模型 ^[71]	69.5	47.3	63.5	66.9	60.6	65.5
GPT2-Medium (Sample)	59.66	31.55	50.63	51.63	47.88	50.83
GPT2-Medium (Beam)	61.85	29.18	50.54	55.32	46.17	53.21
T5-base	74.93	34.35	60.61	74.11	46.35	66.18
综合策略 (Ours)	69.43	47.20	63.45	68.09	63.13	67.04

我们进一步采用当前广泛使用的预训练语言模型 GPT-2^[62] 以及 T5^[63] 作为参照的基线模型。我们发现 T5 模型也能达到不错的语法纠错效果，尤其是在 BEA19 数据上取得了很高的准确性，证明了 T5 预训练模型在语法纠错任务上的有效性。但是，相比较综合语料增广策略以及基于编辑隐状态的生成模型，其效果稍差。上述实验结果进一步证明当前的针对于语法纠错任务的语料增广策略能够很好地训练模型，从而实现比通用预训练语言模型更好的效果。因此，在后续实验中，我们将综合语料增广策略^[19] 训练的语法纠错模型作为我们的基线语法纠错模型。

表 2.6 加入 Lang8 数据对不同语法纠错模型的影响。

Model	CoNLL-2014 (M ²)			BEA19		
	P	R	F _{0.5}	P	R	F _{0.5}
T5-base	75.82	31.04	58.85	71.99	48.94	65.79
T5-base (加入 Lang8 数据)	74.93	34.35	60.61	74.11	46.35	66.18
基线语法纠错模型 (Ours)	69.43	47.20	63.45	68.09	63.13	67.04
基线语法纠错模型 (加入 Lang8 数据)	62.87	46.01	58.58	58.42	60.87	58.90

2.6.3 英文学习者网站数据 Lang8 对于语法纠错效果影响

我们进一步的实验发现在语法纠错模型微调阶段加入英文学习者网站数据 Lang8 后不同的语法纠错语言模型展示了不同的效果。我们以通用预训练语言模型 T5 以及面向语法纠错的综合语料增广策略的预训练模型为例进行实验，实验结果如表2.6所示。

我们发现，两种预训练方法在加入 Lang8 数据后显示了截然不同的效果。在训练 T5 模型时候加入 Lang8 数据，其语法纠错效果会相应提升，然而在面向语法纠错的综合语料增广策略的预训练模型的微调阶段加入 Lang8 数据，其语法纠错效果相应下降。这个现象显示了 Lang8 数据的特点。首先，其中一些标注者能够提供一些对语法纠错模型有价值的语法纠错标注。因此，在对于语法纠错不熟悉的预训练语言模型 T5 上会发挥其作用。但是由于标注者的英文水平不同，因此会产生大量的噪音，这样的噪音会对一些针对语法纠错任务本身进行预训练的模型产生误导。

表 2.7 不同 Lang8 数据筛选策略对语法纠错模型的效果。

模型	CoNLL-2014 (M ²)			BEA19		
	P	R	F _{0.5}	P	R	F _{0.5}
基线语法纠错模型 (Ours)	69.43	47.20	63.45	68.09	63.13	67.04
加入全部 Lang8 数据	62.87	46.01	58.58	58.42	60.87	58.90
随机加入部分 Lang8 数据	69.56	44.75	62.61	68.75	60.25	66.86
基于 BERT 的 Lang8 数据筛选	68.76	45.99	62.56	69.14	61.06	67.36
基于强化学习的 Lang8 数据筛选	71.51	44.94	63.95	70.76	61.14	68.60

表 2.8 基于强化学习的数据筛选方法筛选出数据在文本的修改操作类别上的数量统计。

错误类型	修改操作	# 被选择数据	# 全部数据	比例
Unnecessary	删除	165,213	421,484	39.20%
Replacement	替换	721,642	2,001,100	36.06%
Missing	添加	349,894	1,049,148	33.35%

2.6.4 通过筛选有噪音语料库提升语法纠错效果

在本实验中，我们进一步评测不同数据筛选策略对语法纠错模型训练的影响。首先我们比较了三种数据筛选策略，第一种为随机挑选，即随机挑选出一定数据并与人工标注数据整合，以此来微调我们的基线语法纠错模型。第二种策略为基

于 BERT 的数据筛选策略，在该策略中，我们通过对 BERT 模型进行训练，使得 BERT 模型能够将 Lang8 数据和标注数据进行分类。第三种筛选策略是我们在本章中所介绍的基于强化学习的数据筛选策略。三种策略从 Lang8 数据筛选了同样多的数据（约 40 万句对）并与现有的标注数据（NUCLE，FCE 以及 BEA19 的训练集部分）进一步合并，以此来微调我们的基线语法纠错模型。

表 2.9 基于强化学习的数据筛选方法筛选出数据所保留的语法错误类别数量统计。

错误类型	# 被选择数据	# 全部数据	比例
连词 (CONJ)	12,784	31,423	40.68%
小品词 (PART)	6,069	15,531	39.08%
介词 (PREP)	95,014	247,655	38.37%
副词 (ADV)	38,572	105,046	36.72%
名词 (NOUN)	111,774	307,144	36.39%
词形 (MORPH)	18,444	50,853	36.27%
动词 (VERB)	187,685	523,064	35.88%
限定词 (DET)	132,722	370,266	35.85%
形容词 (ADJ)	18,286	51,093	35.79%
词序 (WO)	12,482	35,015	35.65%
其他 (OTHER)	402,746	1,138,091	35.39%
标点 (PUNCT)	83,170	236,572	35.16%
拼写 (SPELL)	40,400	115,565	34.96%
代词 (PRON)	29,842	86,011	34.70%
正确拼字 (ORTH)	39,356	128,390	30.65%
缩略语 (CONTR)	7,403	30,013	24.67%

如表2.7所示，我们展示了利用不同的数据筛选策略对我们基线语法纠错模型进行微调后在语法纠错任务上的效果。相比较加入全部 Lang8 数据的基线模型，我们发现适当地缩小加入 Lang8 数据的比例可以有效减少 Lang8 数据对于语法纠错模型的影响，但是相比较我们的基线语法纠错模型，其效果还是有所下降。基于强化学习的数据筛选策略在三种数据筛选策略中表现最佳，证明了该模型的有效性。其效果的提升主要来源于语法纠错结果的准确度上的提升。同时我们的实验也证明了 Lang8 数据在语法纠错任务中的价值，为进一步提升语法纠错效果提供了可能。

随后，我们对筛选出来的数据的语法纠错的修改操作，即删除、替换、添加三种操作进一步分析，如表2.8所示。我们发现在三种语法错误修改操作中，添加操

作相对整个 Lang8 数据来说所占比例相对最少，我们猜测由于标注者在对语法错误进行修正的时候，其添加内容往往是多样的，因此，对于添加操作，语法纠错的标准更加不统一，导致了此种语法纠错的修改操作类型在选择出来的数据中占比相对较少。此外，我们发现由于 Lang8 数据来源是语言学习者网站，因此，标注者在修改句子时常常会混入额外的注释，例如：对于给定的句子 “I have a fiancée and she is finding her job .”，标注者将其修正为 “I have a fiancée and she is finding a job . (if she doesn ’t have a job yet use ’ a ’ job , but if she has a job then use ’ her ’ job)”。标注者给定的注释同样属于添加操作。

最后，我们对于筛选出的数据的语法错误类型比例进行了展示，如表 2.9 所示。在其中正确拼字 (ORTH) 以及缩略语 (CONTR) 两种语法错误类型所占比例相对较小。这个现象进一步说明由于 Lang8 数据的标注标准不统一导致了用户在 Lang8 数据的标注过程中产生了不同的标注结果，这些不一致的标注结果会对模型的效果产生负面影响。因此，这个现象提示了我们在构建语法纠错语料库时应当规定合适的标注标准，进一步避免产生不一致的语法纠错标注从而误导语法纠错模型。

2.7 本章小结

本章主要介绍了基于生成模型的语法纠错模型的框架，以及一些广泛使用的语法纠错模型的预训练方法。本文比较了不同的语法纠错模型，对其效果进行了评估，同时通过调整训练策略进一步实现了当前最好的基于生成的语法纠错模型。最后，我们探讨了在语法纠错模型微调过程中来自语言学习者网站的 Lang8 数据集的作用。我们发现这部分数据包含一定的噪音，通过我们设计的语法纠错数据筛选策略能够过滤掉部分噪音并进一步提升语法纠错的性能。

第3章 融合多语法纠错证据的英文语法校对

我们在上一章节详细介绍了语法纠错模型的框架以及如何更好地训练语法纠错模型。但由于许多语法纠错模型都是通过随机替换词语的策略来实现语料增广以及模型训练，导致当前语法纠错模型不可避免地产生错误的语法修改或者忽略掉一些语法错误。因此，之前工作利用语法纠错质量评估模型进一步地确保语法纠错结果的准确性，以此来提升语法纠错的效果。这些训练很好的语法纠错模型可以通过柱搜索解码，进一步生成一些可能的语法纠错假设。我们在实验中发现这些由语法纠错模型生成的语法纠错结果可以提供高质量的语法纠错证据，辅助提升语法错误检查以及语法纠错质量评估的效果，进而能够提升语法纠错的效果。然而，现有的语法错误检查以及语法纠错质量评估模型忽略了来自语法纠错模型所提供的潜在的语法纠错证据，从而限制了模型的能力。

针对此问题，我们在本章中^①提出了基于多语法纠错假设的用于语法错误检查以及语法纠错质量评估的神经验证网络 (VERNet)。我们的模型通过将每一个输入句子和语法纠错假设进行拼接，并进一步建立起语法纠错假设之间的交互，从而实现一个推理图来帮助每一个输入句子和语法纠错假设获取其他可能的语法纠错证据。我们的模型设计了两种注意力机制来传播语法纠错证据，以此来提示可能的语法错误修改方案，从而辅助对输入句子的语法错误检查任务以及对所生成的语法纠错假设的质量评估任务。在四个通用语法纠错数据集上的实验结果表明，我们的 VERNet 模型不仅在语法错误检查上可以达到当前最好的性能，而且可以提供当前最好的语法纠错质量评估结果，并且可以通过对语法纠错模型柱搜索的结果重新排序来进一步提升语法纠错的性能。

3.1 问题描述

语法纠错系统主要是为用户提供文本的语法校对服务，这些系统被期望能够对语法错误进行检测，并且提供精确的修改。

现有的语法纠错系统通常采用序列到序列的生成框架 (Sequence-to-Sequence, Seq2Seq)^[42]，以纠正语法错误或提高句子流畅性。然而，如前一章所示，当前语法纠错的相关工作通常将语法纠错看做一个低资源的生成任务，并通过随机增加、

^① 本章主要工作以“Neural Quality Estimation with Multiple Hypotheses for Grammatical Error Correction”为题发表在 2021 年的国际学术会议 The 2021 Conference of the North American Chapter of the Association for Computational Linguistics - Human Language Technologies (NAACL-HLT 2021) 上。

表 3.1 语法改错模型柱搜索解码得到的语法改错结果。

输入句子: Do one who suffered from this disease keep it a secret of infrom their relatives?
Rank@1: <i>Does someone</i> who suffered from this disease keep it a secret <i>from</i> their relatives?
Rank@2: <i>Does anyone</i> who <i>suffers</i> from this disease keep it a secret <i>from</i> their relatives?
Rank@3: <i>Does anyone</i> who suffered from this disease keep it a secret <i>from</i> their relatives?
Rank@4: <i>Does one</i> who suffered from this disease keep it a secret <i>from</i> their relatives?
Rank@5: <i>Does someone</i> who <i>suffers</i> from this disease keep it a secret <i>from</i> their relatives?

替换和删除等操作生成包含语法错误的句子，以此构建弱监督语料库来对语法改错模型进行充分地训练。然而，应用此种方式训练的语法改错模型往往会导致对语法错误的过分修改或者欠修改^[76]，从而降低了语法改错模型的性能。

如表3.1所示，当前的基于 Seq2Seq 的语法改错系统往往使用柱搜索解码 (Beam Search Decoding)^[70,77] 来产生 K 个最佳的语法改错结果。为了产生更准确的语法改错结果，一些语法改错方面的工作通过对柱搜索结果^[19,61] 或者多个语法改错模型所生成的改错结果^[46] 进行重排序来提升语法改错效果。这些模型通常利用多种语法改错特征来对语法改错结果进行重排序，进而提升语法改错效果。这些特征包括：编辑距离，语言模型打分^[21,46,61,78-79] 以及通过质量评估模型进一步预测语法改错结果的评价指标 $F_{0.5}$ 得分^[80] 等。

来自柱搜索的前 K 个语法改错结果往往有着细微的差别，这些差别通常来自生成模型的不确定性 (Uncertainty)^[81]。这些语法改错结果的不确定性主要来自于语言的多样性、潜在的语言的歧义性以及生成模型的置信度。生成模型所产生的多个解码结果已经被证明可以用来提高机器翻译的性能^[82]。Fomicheva 等人^[83] 进一步在机器翻译领域利用机器翻译模型产生的多个解码结果来提升对机器翻译结果的翻译质量评价，并证明此种融合多个机器翻译的解码结果的机器翻译评价方法与人类评价结果更加相关，从而说明了生成模型所提供的多个解码结果对于质量评估的重要意义。

类比于机器翻译的相关工作^[17,42,44,70,82-83]，语法改错任务同样应用了生成模型对语法错误进行修正以及柱搜索进行解码。通过柱搜索解码得到的若干个语法改错结果可以提供有价值的语法改错证据，进而辅助语法改错质量评估过程。

- 柱搜索解码可以提供更好的语法改错结果。对于已有的语法改错模型^[19,22]，在柱搜索解码中排名第一的语法改错结果和其中最好的语法改错结果之间的语法改错性能还有很大的差距。我们在 CoNLL-2014 数据集上，对 Zhao 等人提出的语法改错模型^[22] 和 Kiyono 等人提出的语法改错模型^[19] 进行了评测，这些系统的 $F_{0.5}$ 分数分别为 58.99 和 62.03，但是，这些系统柱搜索解码

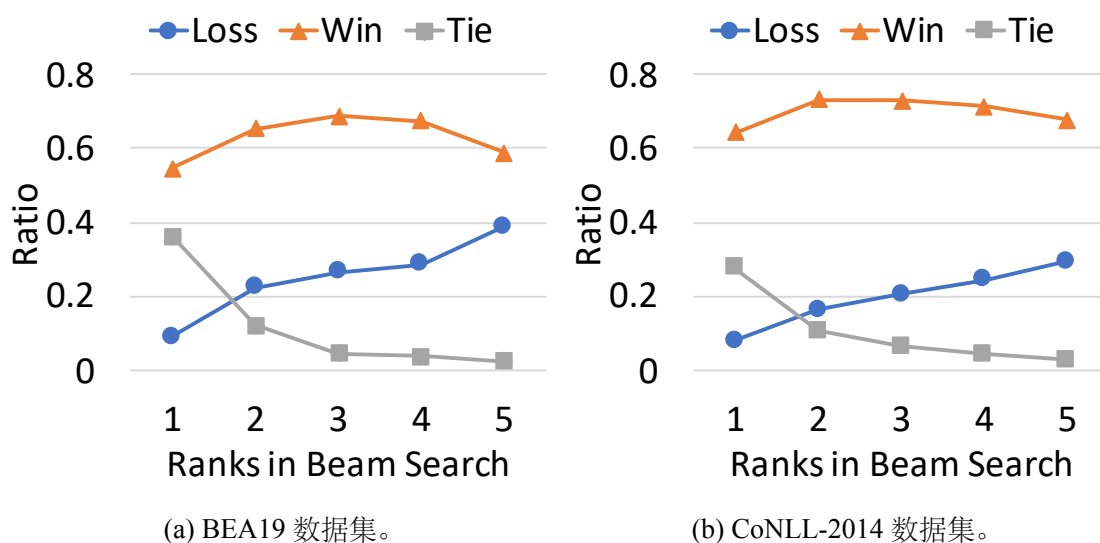


图 3.1 柱搜索解码所生成的语法纠错结果的语法合理性。

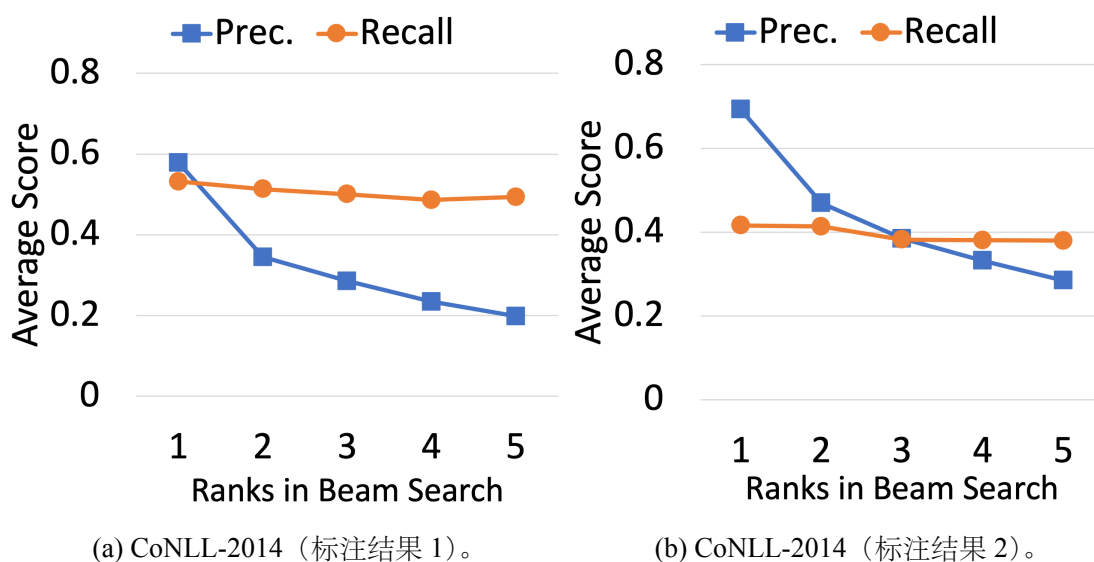


图 3.2 柱搜索解码所生成的语法纠错结果的语法纠错质量。

中给出的最佳语法结果 $F_{0.5}$ 分数可以达到 73.56 和 76.82。这个现象证明了柱搜索解码可以提供更好的语法纠错结果。

- **柱搜索解码得到的语法纠错结果更加具有语法合理性。**为了评估语法纠错结果的语法合理性，我们采用了 BERT 语言模型来对柱搜索解码结果进行打分。首先我们采用了当前最先进的基于生成的语法纠错模型^[19]生成若干个语法纠错结果，并展示了在不同柱搜索排序（从第一名到第五名）中语法纠错结果的语言模型得分情况。我们将每一个相应排名的语法纠错结果与输入句子进行比较，并通过基于 BERT 的语言模型对二者进行打分，并将语法纠错结果分为三种类型，Win（语法纠错模型给出的结果更好），Tie（语法纠错结果

和输入句子相同) 和 Loss (输入句子比语法纠错结果更好), 然后将这三种类型所占的比例在图中画出。在这里我们采用两个通用语法纠错评测数据集 BEA19 和 CoNLL-2014 来测试语法纠错模型在柱搜索解码中提供的语法纠错结果的语法合理性。如图3.1所示, 经过良好训练的语法纠错模型通过柱搜索解码生成的语法纠错结果相比较输入句子通常会赢得语言模型更多的青睐, 即使这些语法纠错结果的排名相对靠后。这个现象说明了这些语法纠错结果通常比输入句子更符合语法要求。

- **柱搜索解码得到的语法纠错结果可以提供有价值的语法纠错证据。**我们进一步评价当前最先进的基于生成的语法纠错模型^[19]通过柱搜索解码所生成的语法纠错结果的质量。我们将所得到的语法纠错结果分别根据 CoNLL-2014 的两个标注结果来计算在不同柱搜索排序位置上的语法纠错结果的语法纠错准确度 (Precision) 和召回率 (Recall) 的平均分数。如图3.2所示, 具有不同柱搜索排名的语法纠错结果具有几乎相同的平均召回率得分, 这表明在柱搜索解码中的所有语法纠错假设都有可能提供一些有价值的语法纠错证据。排序在后面的语法纠错结果往往提出了冗余或错误的语法错误修正方案, 因此降低了语法纠错的性能。

然而, 以往的语法纠错质量评估模型^[80]以及语法错误检查模型^[9-10,21], 忽略了语法纠错模型所提供的多个语法纠错结果中的高质量语法纠错证据, 从而限制了语法错误检查模型以及语法纠错质量评估模型的效果。因此, 我们希望在语法错误校对方面能够充分利用语法纠错模型所提供的多个纠错结果, 以此来提示模型可能的语法错误以及修正结果, 进一步提升语法错误检查以及语法纠错质量评估的效果, 并通过对柱搜索结果进行重排序来提升语法纠错的效果。

针对此问题, 我们提出了一种融合多语法纠错证据的英文语法校对模型 (Neural Verification Network, VERNet), 并通过建立多条语法纠错证据的交互, 进一步从不同语法纠错结果中获取语法纠错证据, 从而对输入句子的语法错误进行检查以及对语法纠错的质量进行评估。

总结来说我们的工作有如下三个创新点:

(1) 我们发现现有的语法纠错模型在柱搜索解码过程中产生的语法纠错结果可以提供高质量的语法纠错证据, 因此可以通过整合多个语法纠错结果中的纠错证据来提升语法错误检查、语法纠错质量评估的效果。

(2) VERNet 将〈输入句子、语法纠错句子〉对进行拼接并作为一个节点, 并连通所有的节点, 从而构造一个完全连通的推理图。该推理图建立了语法纠错结果之间的交互, 从而能够帮助每两个节点之间传播相关的语法纠错证据。然后, VERNet

提出了两种注意力机制：节点交互注意力机制（Node Interaction Attention）以及节点选择注意力机制（Node Selection Attention），从而能够从其他语法纠错结果中提取和总结必要的语法纠错证据。

(3) 我们的实验表明，VERNet 能够从若干个语法纠错结果中提取出有意义的语法纠错证据，从而帮助提升语法错误检查的效果，提升语法纠错质量评估的能力，进而提升语法纠错模型的性能。相比较其他基线模型，VERNet 达到了当前最好的语法错误检查以及语法纠错质量评估的效果。我们进一步的分析发现 VERNet 能够帮助语法纠错模型在几乎所有的语法错误类型上取得更好的效果，进一步证明了其有效性。

3.2 相关工作

语法纠错任务是为语法层面的校对设计的，希望能够自动化地识别句子中的语法错误并进行合适的修改。大规模的人工标注语料库的构建^[35-37]为完全数据驱动的语法纠错模型提供了基础。

现有的语法纠错工作大多将语法纠错任务视为一个自然语言生成（Natural Language Generation, NLG）任务，通常使用序列到序列结构^[42]通过柱搜索解码^[45-46]生成语法纠错结果。基于 Transformer 的自然语言生成框架^[17]在许多自然语言处理任务中都显示了其有效性，同时也被相关工作用于实现当前最好的语法纠错模型^[19,47]。

对语法纠错模型通过柱搜索解码所得到的前 K 个语法纠错结果或者不同语法纠错模型所提供的多个语法纠错结果进行重新排序^[80,84]，能够更好地帮助语法纠错模型提升其性能。其中一些方法通过评估每一个语法纠错结果是否符合语言学规律和语法规则，进而判断改正后的句子的语法纠错质量。为此，这些工作通常采用预训练语言模型对语法纠错结果进行打分^[46,78]，亦或是利用语法错误检查（Grammatical Error Detection, GED）模型来评估语法纠错的质量。其中语法错误检查模型^[9-10]被用于判断句子含有语法错误的概率，在这里一般通过在句子级别^[79]或者词语级别^[21]的语法错误标签进行训练，随后利用训练好的模型对语法纠错结果进行重排序进一步提升语法纠错效果。

质量评估模型^[85-91]已经被广泛地应用在机器翻译领域，通过预测 BLEU^[92]值进一步地来提升机器翻译的效果。语法纠错质量评估模型^[80]也借鉴了其思路，并根据语法纠错质量评估模型的打分对语法纠错结果进行重排序。该模型通过编码器-解码器架构（Encoder-Decoder）进一步地建立输入文本和语法纠错结果之间的交互，并通过预测语法纠错官方评估分数 $F_{0.5}$ 以评估语法纠错质量。不同于之

前提到的基于语言模型的重排序方法，语法纠错质量评估模型通过考虑了语法纠错的准确性，进一步提升语法纠错的性能。

随着预训练语言模型的发展^[15,50,93]，诸如 BERT^[15] 之类的模型已经证明了它在文本编码以及产生词语上下文表示方面的强大能力，并且被应用在多种自然语言处理任务上。语法错误校对的相关任务也将预训练语言模型 BERT 应用在语法错误校对不同阶段，从而能够进一步提升语法错误检查或者语法纠错效果。例如：采用预训练语言模型 BERT 对语法纠错结果进行打分并重排序^[78-79]，利用预训练语言模型 BERT 提升语法错误检查的效果^[16]，或者通过预训练语言模型 BERT 对输入句子进行编码并得到相应的词语的上下文表示，进一步将输入句子的词语表示作为语法纠错模型编码器（Encoder）端的一部分^[61,94]。

3.3 模型框架

为了充分利用语法纠错模型所提供的语法纠错证据，本小节具体介绍了基于多种语法纠错结果的用于语法错误检查以及语法纠错质量评估的神经验证网络（Neural Verification Network, VERNet）。如图3.3所示，为了利用多个语法纠错结果来验证当前词语（ c_p^k ）的语法纠错质量，我们希望模型能够从多个语法纠错结果中找到潜在的语法纠错支持证据，以此来验证词语 c_p^k 的语法纠错质量，进而提升预训练语言模型在语法校对相关任务上的效果。

当给定一个输入句子 s ，我们可以利用一个语法纠错模型通过柱搜索解码得到 K 个与之相对应的语法纠错结果，我们将其标记为 $C = \{c^1, \dots, c^k, \dots, c^K\}$ 。我们首先将每一个输入句子和纠错结果句子对 $\langle s, c^k \rangle$ 看做一个节点并且全连接所有的节点，以此来建立多个语法纠错结果之间的交互。随后，我们采用预训练语言模型 BERT 进一步得到在输入句子和纠错结果句子对 $\langle s, c^k \rangle$ 中每个词语的表示，并且通过设计的两种注意力机制来提取、传播并整合从其他语法纠错结果中得到的语法纠错证据。我们利用这些语法纠错证据去验证输入句子和纠错结果句子对 $\langle s, c^k \rangle$ 中每一个词语的质量标签（对于输入句子 s 为语法质量标签，对于语法纠错结果 c^k 则为语法纠错质量标签）。最后，我们可以利用 VERNet 模型对语法纠错结果 c^k 中的每一个词语进行打分，进一步对词语的得分进行平均从而得到语法纠错质量评估分数。

3.3.1 句子对表示初始化

诸如 BERT^[15] 之类的预训练语言模型能够通过产生高质量的词语级别的上下文表示来显著提升自然语言处理任务的效果。在这里，对于给定的包含 m 个词语

Source Sentence:

s : Do one who suffered from this disease ...

Hypotheses from the beam search decoding of basic GEC :

c^1 : Does someone who suffered from this disease ...

c^k : Does someone who suffers (p -th token) from this disease ...

c^l : Does anyone who suffers from this disease ...

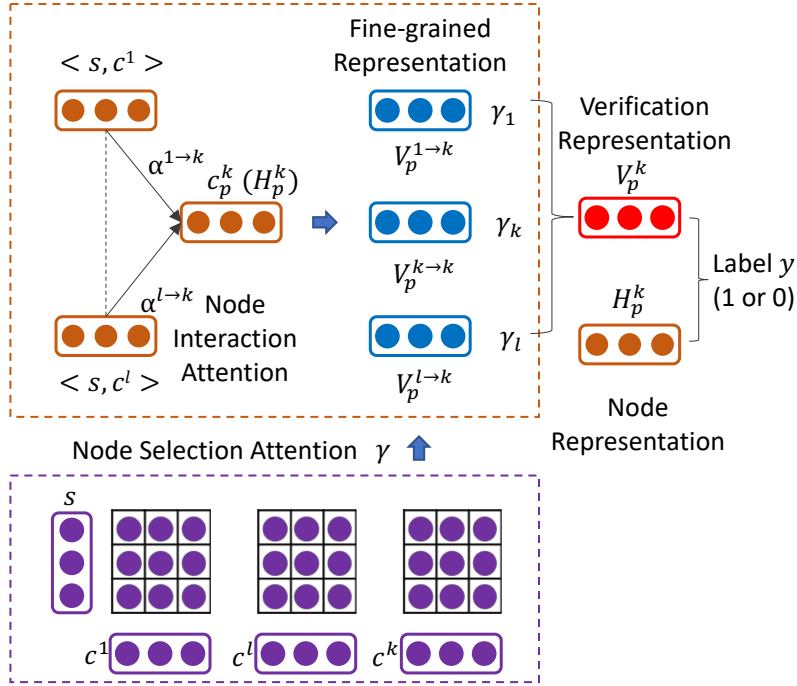


图 3.3 基于多语法纠错假设的神经验证网络 (Neural Verification Network, VERNet) 模型框架图。

的输入句子 s 以及包含 n 个词语的第 k 个语法纠错结果 c^k ，我们将输入句子和纠错结果句子对 $\langle s, c^k \rangle$ 利用表示句子开头和结尾的特殊符号 “[CLS]” 以及 “[SEP]” 进行拼接，然后使用 BERT 进行编码，并得到节点初始化表示 H^k ：

$$H^k = \text{BERT}([\text{CLS}]s[\text{SEP}]c^k[\text{SEP}]). \quad (3-1)$$

输入句子和纠错结果句子对 $\langle s, c^k \rangle$ 的节点初始化表示 H^k 包含了每个词语的隐状态表示，我们将其标记为 $H^k = \{H_0^k, \dots, H_{m+n+2}^k\}$ 。

3.3.2 通过多语法纠错结果验证词语质量

VERNet 设计了两种注意力机制，即节点交互注意力机制 (node interaction attention) 和节点选择注意力机制 (node selection attention)，从多个语法纠错结果中获取能够验证第 k 个节点中每个词语的支持证据，并进一步形成能够验证第 k 个节点词语质量的节点验证表示 V^k 。

其中节点交互注意力机制能够从第 l 个节点中总结有价值的语法纠错证据，并

形成其节点的细粒度表示 $V^{l \rightarrow k}$ 。随后，节点选择注意力机制根据每一个节点的置信度计算出每一个节点的权重 γ^l ，进一步通过该权重整合节点的细粒度表示 $V^{l \rightarrow k}$ 并计算出第 k 个节点的节点验证表示 V^k 。

基于节点交互注意力机制的细粒度节点表示。 VERNet 模型所设计的节点交互注意力机制通过计算注意力权重 $\alpha^{l \rightarrow k}$ 有选择地从第 l 个节点中读取词语，进一步从其中选择出能够对第 k 个节点中的词语起到支持作用的语法纠错证据，从而形成节点的细粒度表示 $V^{l \rightarrow k}$ 。

对于第 k 个节点中的第 p 个词语 w_p^k ，我们首先依据词语 w_p^k 和第 l 个节点中的第 q 个词语 w_q^l 的词语表示之间的相关性分数，计算出节点交互注意力机制的权重 $\alpha_q^{l \rightarrow k}$ ：

$$\alpha_q^{l \rightarrow k} = \text{softmax}_q((H_p^k)^T \cdot W \cdot H_q^l), \quad (3-2)$$

这里 W 是一个可训练参数。 H_p^k 和 H_q^l 分别为两个词语 w_p^k 和 w_q^l 的向量化表示。随后，我们利用节点交互注意力权重 $\alpha_q^{l \rightarrow k}$ 对第 l 个节点中所有词语的表示进行加权求和，从而得到对于第 k 个节点中的第 p 个词语 w_p^k 来说，第 l 个节点的细粒度节点表示 $V_p^{l \rightarrow k}$ ：

$$V_p^{l \rightarrow k} = \sum_{q=1}^{m+n+2} (\alpha_q^{l \rightarrow k} \cdot H_q^l). \quad (3-3)$$

进一步，通过 $V_p^{l \rightarrow k}$ ，我们可以得到面向第 k 个节点的第 l 个节点的细粒度节点表示 $V^{l \rightarrow k} = \{V_1^{l \rightarrow k}, \dots, V_p^{l \rightarrow k}, \dots, V_{m+n+2}^{l \rightarrow k}\}$ 。

通过节点选择注意力机制整合语法纠错证据。 VERNet 设计了节点选择注意力机制通过计算节点的重要程度进一步地从多个节点中整合能够支持验证当前词语质量的语法纠错证据。具体来说，对于第 l 个节点，我们通过 Attention-over-attention 机制^[95] 来对节点中的输入句子 s 和语法纠错结果 c^l 中的词语分别进行加权表示，从而计算出对于第 l 个节点的节点选择注意力权重 γ^l 。随后，我们利用所得到的节点选择注意力权重 γ^l 对第 l 个节点的细粒度节点表示 $V^{l \rightarrow k}$ 进行加权求和，最终得到面向第 k 个节点的节点验证表示 V_p^k 。

为了计算节点的选择注意力权重 γ^l ，我们首先在第 l 个节点中的输入句子和纠错结果两者之间建立交互矩阵 M^l 。在交互矩阵 M^l 中的每一个元素 M_{ij}^l 衡量了第 l 个节点中输入句子的第 i 个词语以及语法纠错结果中的第 j 个词的相关程度，在这里，用于标记句子结尾的结束符 “[SEP]” 也被纳入到计算当中来：

$$M_{ij}^l = (H_i^l)^T \cdot W \cdot H_{m+1+j}^l, \quad (3-4)$$

这里 W 是一个可训练参数，我们随后在输入句子和语法纠错结果两个维度上，利

用 Attention-over-attention 机制计算出相应的注意力权重 β_i^{ls} 和 β_j^{lh} :

$$\beta_i^{ls} = \frac{1}{n+1} \sum_{j=1}^{n+1} \text{softmax}_i(M_{ij}^l), \quad (3-5)$$

$$\beta_j^{lh} = \frac{1}{m+1} \sum_{i=1}^{m+1} \text{softmax}_j(M_{ij}^l). \quad (3-6)$$

随后我们利用两组注意力权重 β_i^{ls} 和 β_j^{lh} , 进一步得到第 l 个节点中输入句子和改错结果的表示 h^{ls} 以及 h^{lh} :

$$h^{ls} = \sum_{i=1}^{m+1} \beta_i^{ls} \cdot H_i^l, \quad (3-7)$$

$$h^{lh} = \sum_{j=1}^{n+1} \beta_j^{lh} \cdot H_{m+1+j}^l. \quad (3-8)$$

最后, 对于第 l 个节点的节点选择注意力权重 γ^l 被计算出来, 并用来对语法改错证据线索进行加权整合:

$$\gamma^l = \text{softmax}_l(\text{Linear}((h^{ls} \circ h^{lh}); h^{ls}; h^{lh})), \quad (3-9)$$

这里 \circ 代表着元素对应位置相乘操作 (element-wise multiplication), 在这里符号 \circ 代表着拼接操作。

节点选择注意力权重 γ^l 进一步地整合对于第 k 个节点中的第 p 个词语 w_p^k 的第 l 个节点的细粒度节点表示 $V_p^{l \rightarrow k}$ 并形成面向第 k 个节点中的第 p 个词语 w_p^k 的节点验证表示 V_p^k :

$$V_p^k = \sum_{l=1}^K (\gamma^l \cdot V_p^{l \rightarrow k}), \quad (3-10)$$

我们进一步可以得到面向第 k 个节点的节点验证表示 $V^k = \{V_1^k, \dots, V_p^k, \dots, V_{m+n+2}^k\}$ 。

3.3.3 语法改错质量评估

对于第 k 个节点中第 p 个词语 w_p^k , 我们可以利用相应的节点验证表示 V_p^k 来计算出词级别的语法改错质量评估标签 y 的概率 $P(y|w_p^k)$:

$$P(y|w_p^k) = \text{softmax}_y(\text{Linear}((H_p^k \circ V_p^k); H_p^k; V_p^k)), \quad (3-11)$$

这里 \circ 代表着元素对应位置相乘操作 (element-wise multiplication), ; 则是代表着拼接操作。我们进一步对第 k 个节点中的语法改错结果中的全部词语质量评估分

数的得分（即概率 $P(y = 1|w_p^k)$ ）进行平均，从而得到对于输入句子和纠错结果句子对 $\langle s, c^k \rangle$ 的语法纠错质量评估分数 $f(s, c^k)$ ：

$$f(s, c^k) = \frac{1}{n+1} \sum_{p=m+2}^{m+n+2} P(y = 1|w_p^k). \quad (3-12)$$

3.3.4 端到端训练

我们进一步在词级别的标签下训练我们的 VERNet 模型。这里我们同时将面向输入句子和面向纠错结果的词级别的训练标签作为监督信号来指导模型的学习，其意义分别为对于输入句子的语法错误的标注以及对于语法纠错结果的准确性的标注。

我们使用交叉熵损失函数（cross entropy loss）来计算对于第 k 个节点中第 p 个词语 w_p^k 的词语质量分类标签 y 的损失：

$$L(w_p^k) = \text{CrossEntropy}(y^*, P(y|w_p^k)), \quad (3-13)$$

这里 y^* 代表着标注数据所给定的真实的标签分类。

随后我们可以计算出 VERNet 训练时候的损失函数：

$$L = \frac{1}{K} \frac{1}{m+n+2} \sum_{k=1}^K \sum_{p=1}^{m+n+2} L(w_p^k). \quad (3-14)$$

3.4 实验

这一节中，我们将介绍在语法错误检查、语法纠错质量评估以及语法纠错三个任务中，实验所用到的数据集，实验设置，基线模型以及最终的实验结果。

表 3.2 VERNet 模型采用的数据信息统计。

数据集	训练集	验证集	测试集
FCE	28,350	2,191	2,695
BEA19	34,308	4,384	4,477
NUCLE	57,151	-	-
CoNLL-2014	-	-	1,312
JFLEG	-	-	747
Total	119,809	6,575	9,231

表 3.3 词语级别的语法错误标注标签生成示例。

输入句子	The ₁ a ₂ Mobile phone is a marvelous invention to ₉ charge ₁₀ the world ₁₂ [SEP]		
语法改错信息	修改操作	被修改部分	修改词语
	删除	1,2	-
	替换	9,10	change
	插入	12,12	.

3.4.1 数据集

在表3.2中, 我们展示了实验中所使用的语法改错数据集的统计信息。在实验中我们不加入额外的训练语料, 以实现模型的公平评测。其中三个数据集 FCE^[65]、BEA19^[37] 和 NUCLE^[36] 被用来构建训练集和开发集。

为了构建适用于训练 VERNet 的训练集、开发集以及测试集, 我们使用自动化语法错误信息标注工具 ERRANT^[96-97] 来对输入句子和语法改错模型所生成的语法改错结果句子, 参照人工标注者所给定的语法改错结果进行标注, 并且标注需要对句子进行修改的地方。自动化语法改错信息标注工具包 ERRANT 会针对给定句子通过进行多种编辑操作, 如: 删除, 插入和替换, 进一步得到我们所期望的语法改错结果。因此, 我们对输入句子以及语法改错模型所生成的改错结果与人工标注结果利用 ERRANT 进行标记后得到了序列标注标签, 其分别表示了输入句子词语的语法正确性和语法改错模型所提供的语法改错结果的准确性。其中每个词语都被标记为正确 (即标记标签为 1) 或不正确 (即标记标签为 0)。如表3.3所示, 在该示例句子中, 被标记为红色的词语代表了其为不正确 (即标记为标签 0) 而其他的词语则标记为正确 (即标记为标签 1)。其中 “[SEP]” 特殊符号用来表示句子的结尾符。

接下来我们分别对英文语法校对的三个任务进行评测, 分别为语法错误检查 (Grammatical Error Detection, GED)、语法改错质量评估 (Quality Estimation, QE) 以及语法改错 (Grammatical Error Correction, GEC)。对于语法错误检查任务, 我们采用与之前工作^[9-10,21] 一致的评价设置, 采用准确率 (Precision), 召回率 (Recall), 和 $F_{0.5}$ 对每个词语的语法正确性预测效果进行评价。在语法错误检查任务中, 语法改错数据 FCE^[65] 和 CoNLL-2014^[8] 被用来评测模型效果。另外, 四个通用语法改错测试数据集, FCE^[65], BEA19 (Restrict)^[37], CoNLL-2014^[8] 和 JFLEG^[66], 被用来评估模型在语法改错质量评估任务以及语法改错任务上的效果。对于 FCE, BEA19 以及 CoNLL-2014 我们依旧采用准确率 (Precision), 召回率 (Recall), 和 $F_{0.5}$ 指标对模型效果进行评价^[96,98]。对于 JFLEG 我们则采用官方评价指标 GLEU^[99]

进行评测。此外，为了验证语法改错质量评估的效果，我们也沿用了之前语法改错质量评估工作^[80]中对模型效果的评价指标，皮尔逊相关系数（Pearson Correlation Coefficient, PCC）^[100]，来评价不同语法改错质量评估模型的打分与语法改错结果的真实 $F_{0.5}$ 得分之间的相关性。

3.4.2 基线模型

我们将本章所提出的模型分别在语法改错、语法错误检查、以及语法改错质量评估三个任务上与不同的基线模型进行比较。

对于语法改错任务我们比较了如下模型：

Basic GEC Model^[19] 是当前最好的基于生成模型的语法改错模型，我们将其作为我们主要的基线模型。

BERT fuse (GED)^[61] 在语法错误检查任务上训练预训练语言模型 BERT，强化其对语法错误的识别能力，并将通过 BERT 得到的输入句子的表示融合到我们的基线语法改错模型（Basic GEC Model）的编码器一端。

对于语法错误检查任务我们对比了如下模型：

LSTM^[9-10] 包含了三个基于长短期记忆循环神经网络（Long Short Term Memory, LSTM）^[14] 作为编码器的语法错误检查模型 LSTM, BiLSTM-ATTN 以及 BiLSTM-JOINT。LSTM 通过引入额外的语言模型训练目标进一步提升语法错误检查的效果。BiLSTM-ATTN 以及 BiLSTM-JOINT 两个模型则是在 LSTM 模型的基础上通过设计不同的注意力机制，引入对于注意力机制的限制以及句子级别的监督信号来进一步提升模型在语法错误检查上的效果。

BERT-GED (SRC)^[16] 利用预训练语言模型 BERT 对输入文本进行编码进一步实现了语法错误检查模型，该模型达到了在当前语法错误检查任务中的最好水平。此外，另外一种考虑语法改错结果的语法错误检查模型 BERT-GED (JOINT) 也被作为基线模型在语法错误检查任务中进行了比较。

对于语法改错质量评估任务我们对比了如下模型：

NQE^[80] 使用编码器-解码器结构对输入句子和改错句子进行编码，并设计了预测器预测语法改错官方评价指标 $F_{0.5}$ 分数。该模型应用不同的神经网络架构，如循环神经网络（Recurrent Neural Network, RNN）和卷积神经网络（Convolutional Neural Network, CNN）来实现编码器-解码器以及预测器。NQE 模型有四个变种模型 NQE (CC)、NQE (RC)、NQE (CR) 以及 NQE (RR)。对于 NQE (XY)，X 表示编码器-解码器结构，Y 表示预测器结构。X 和 Y 可以是循环神经网络 (R) 或卷积神经网络 (C)。

此外，我们还参考了一些相关工作利用预训练语言模型 BERT 实现如下基线

模型来验证 VERNet 模型在语法纠错质量评估任务上的效果。我们进一步给出各个模型对于语法纠错质量评估分数的计算过程。

首先给定了包含 m 个词语的输入句子 s 和包含 n 个词的第 k 个语法纠错结果 c^k ，我们可以通过 BERT 得到句对 $\langle s, c^k \rangle$ 的表示 H^k ：

$$H^k = \text{BERT}([\text{CLS}] s [\text{SEP}] c^k [\text{SEP}]), \quad (3-15)$$

或者是通过 BERT 获得针对于第 k 个语法纠错结果 c^k 的表示 \mathcal{H}^k ：

$$\mathcal{H}^k = \text{BERT}([\text{CLS}] c^k [\text{SEP}]). \quad (3-16)$$

其中 “[CLS]” 和 “[SEP]” 代表了特殊标识符。“[CLS]” 标识符的表示 H_0^k 和 \mathcal{H}_0^k 被作为输入句对或句子的表示。

BERT-LM^[78] 通过对第 k 个语法纠错结果 c^k 中的每个词语进行掩码，进而计算出 c^k 的语言模型困惑度 (Perplexity, PPL)，以此来衡量语法纠错效果：

$$f_{\text{LM}}(c^k) = -\text{PPL}(\mathcal{H}_{1:n}^k), \quad (3-17)$$

这里困惑度越小则代表句子的纠错结果更好。

BERT-GQE^[79] 利用第 k 个语法纠错结果的 “[CLS]” 的隐状态表示 \mathcal{H}_0^k 进一步预测句子有无语法错误的概率 $P(y_s | c^k)$ ，以此来衡量语法纠错的效果：

$$P(y_s | c^k) = \text{softmax}_{y_s}(W \cdot \mathcal{H}_0^k), \quad (3-18)$$

这里 W 是一个可学习参数。 y_s 标记了进行语法纠错后的句子是否有语法错误，其中无语法错误被标记为 1 ($y_s = 1$)，反之标记为 0 ($y_s = 0$)。

随后语法纠错结果 c^k 的质量评估分数即可计算出来：

$$f_{\text{GQE}}(c^k) = P(y_s = 1 | c^k). \quad (3-19)$$

BERT-QE 类似于 NQE^[80]，其使用输入句子 s 和第 k 个语法纠错结果 c^k 的 “[CLS]” 的隐状态表示 H_0^k 预测质量评估分数：

$$f_{\text{QE}}(s, c^k) = \text{sigmoid}(W \cdot H_0^k), \quad (3-20)$$

这里 W 是一个可学习参数。我们使用 BERT-QE 预测第 k 个语法纠错结果 c^k 的语法纠错质量评估分数 $f_{\text{QE}}(s, c^k)$ ，并训练 BERT-QE 模型使其预测的分数接近于 c^k 的 $F_{0.5}$ 分数。

BERT-GED 包含了三个模型 BERT-GED (SRC), BERT-GED (HYP) 以及 BERT-GED (JOINT)。BERT-GED (SRC) 参考了基于 BERT 的语法错误检查模型^[16]来预测词级别语法错误标签，进一步我们参考了前人的工作^[21]利用得到的语法错误检查模型来对语法纠错结果进行质量评估。对于另外两种模型 BERT-GED (HYP) 和

BERT-GED (JOINT), 它们利用 BERT 对输入句子和语法纠错结果进行编码, 同样预测词级别的语法纠错质量评估标签。不同的是 BERT-GED (HYP) 仅仅利用在语法纠错结果上的标注信息作为监督信号对模型进行训练, 而 BERT-GED (JOINT) 同时将在输入句子上的标注信息以及语法纠错结果上的标注信息作为监督信号引入到模型的训练过程中, 从而实现联合训练。

由于我们在 BERT-GED 的三种模型中都采用相同的方式计算句子级别语法纠错质量评估分数, 因此, 在这里我们以 BERT-GED (HYP) 为例来展示如何计算句子的语法纠错质量评估分数。我们首先利用 BERT 得到第 k 个语法纠错结果 c^k 的所有词语隐状态表示 $H_{m+2:m+n+2}^k$ 。类似 VERNet, 在这里我们依旧是将结尾标识符 “[SEP]” 引入到句子的语法纠错质量评估分数计算中。

对于第 k 个语法纠错结果中的第 i 个词语 w_i^k , 我们通过其隐状态 H_{m+1+i}^k 来计算出词语 w_i^k 的语法纠错质量标签 y :

$$P(y|w_i^k) = \text{softmax}(W \cdot H_{m+1+i}^k), \quad (3-21)$$

这里 W 是一个可学习参数。语法纠错质量标签 y 被分为两类: 语法错误修改正确 ($y = 1$) 和语法错误修改错误 ($y = 0$)。对于句子级别的语法纠错质量评估分数, 我们将词级别的语法纠错质量评估分数 $P(y = 1|w_i^k)$ 进行平均, 进一步计算出第 k 个语法纠错结果的句子级别的语法纠错质量评估分数 $f(s, c^k)$:

$$f_{\text{GED}}(s, c^k) = \frac{1}{n+1} \sum_{i=m+2}^{m+n+2} P(y = 1|w_i^k). \quad (3-22)$$

3.4.3 实验设置

为了生成语法纠错结果, 我们使用当前最先进的基于自回归生成模型的语法纠错系统^[19] 作为我们的基线语法纠错模型 (Basic GEC Model), 并与其保持相同的实验设置。我们将基线模型的柱搜索的大小 (Seam Size) 设置为 5^[19], 并在实验中保留了全部的解码结果。

在所有实验中, 我们使用 BERT^[15] 和 ELECTRA^[101] 的基本版本来编码输入文本。BERT 是一种广泛使用的预训练语言模型, 其通过掩码语言模型任务进行预训练。ELECTRA 则是采用词语替换检测任务进行模型预训练, 旨在预测词语是否被基于 BERT 的生成器所替换。ELECTRA 是基于判别器的预训练语言模型, 其更贴近于语法错误检查任务。我们将 BERT 作为文本编码的主要模型, 并利用 ELECTRA 评估模型的泛化能力。

我们实验中的 BERT 和 ELECTRA 模型都继承了 Huggingface 的 PyTorch 实现^[74]。此外, 我们采用 Adam^[75] 用于参数优化, 学习率设置为 5e-5, 批处理 (Batch

Size) 大小设置为 8, 并且在训练过程中将模型优化步长 (Accumulate Step) 设置为 4。此外, 我们将输入句子和语法纠错结果的最大句子长度分别设置为 120。

在对语法纠错模型柱搜索解码所得到的语法纠错重排序阶段, 我们采用了 Learning-to-rank 中的坐标上升方法 (Coordinate Ascent, CA)^[102] 来整合语法纠错质量评估的得分以及基线语法纠错模型所给定的得分并计算最终分数, 以此来对柱搜索解码得到的语法纠错结果进行重排序。在坐标上升方法中, 我们通过将 $F_{0.5}$ 得分最高的语法纠错结果作为正例, 将其他低于该得分的语法纠错结果作为负例进一步训练并得到对语法纠错质量评估模型以及基线语法纠错模型分数的加权重。坐标上升方法由 RankLib^①实现。

表 3.4 基于 BERT 和长短期记忆循环神经网络 (LSTM) 的语法错误检查模型在 FCE 数据集^[65]上的效果。

模型	P	R	$F_{0.5}$
LSTM ^[9]	58.88	28.92	48.48
BiLSTM-ATTN ^[10]	60.73	22.33	45.07
BiLSTM-JOINT ^[10]	65.53	28.61	52.07
BERT-GED (SRC)	73.69	45.39	65.52

3.4.4 语法错误检查效果

在此实验中, 我们探究了 VERNet 模型在语法错误检查任务上的效果。

首先, 我们采用了在之前的语法错误检查工作^[9-10]中被广泛使用的数据集 FCE 来评估基于 BERT 和长短期记忆循环神经网络 (LSTM) 的语法错误检查模型的效果。我们保持与以前的工作相同的数据集以及实验设置^[10]。如表 3.4 所示, 我们展示了三个基于长短期记忆循环神经网络^[14]的模型 LSTM^[9], LSTM-ATTN^[10]和 LSTM-JOINT^[10]在标准语法错误检查数据集 FCE 上的效果。由于实验设置相同, 我们从相应论文中将基于长短期记忆循环神经网络的语法错误检查模型的实验结果借用过来。与基于长短期记忆循环神经网络的语法错误检查模型相比, 基于 BERT 的语法错误检查模型有着 13% 的提升, 进一步显示出 BERT 对于语法错误检查的良好效果。因此, 在后面的实验中, 我们将基于 BERT 编码的语法错误检查模型作为我们主要的基线模型。

进一步, 我们对基于 BERT 模型编码的不同语法错误检查模型的效果进行评估, 其结果如表 3.5 所示。在实验中, 两种基线模型 BERT-GED (SRC) 以及 BERT-GED (JOINT) 被比较。其中, BERT-GED (SRC) 不考虑任何语法纠错模型所提供的

① <https://sourceforge.net/p/lemur/wiki/RankLib/>

表 3.5 基于 BERT 的不同语法错误检查模型的效果。

模型	FCE test set			CoNLL-2014 ann. 1			CoNLL-2014 ann. 2		
	P	R	F _{0.5}	P	R	F _{0.5}	P	R	F _{0.5}
BERT-GED (SRC)	74.22	43.34	64.97	59.84	27.11	48.20	77.94	25.02	54.77
BERT-GED (JOINT)	75.62	44.44	66.32	60.79	27.33	48.83	77.42	25.23	54.77
BERT-VERNet	81.53	45.71	70.48	62.64	30.62	51.80	82.25	28.49	59.71
ELECTRA-VERNet	80.94	50.51	72.24	62.50	35.61	54.30	81.69	32.97	63.06

语法纠错结果。BERT-GED (JOINT) 仅考虑基线语法纠错模型所提供的排名最高的一个语法纠错结果。BERT/ELECTRA-VERNet 考虑了全部的柱搜索解码所得到的语法纠错结果，并采用我们的 VERNet 框架对多个语法纠错结果中的语法纠错证据进行提取。

与 BERT-GED (SRC) 相比，BERT-GED (JOINT) 通过引入语法纠错模型所提供的排名第一的语法纠错结果进一步提升了其语法错误检查性能。这说明语法纠错结果能够对语法错误检查起到提示作用，从而能够辅助提升语法错误检查效果。当我们利用 VERNet 将全部柱搜索解码所得到的语法纠错结果引入到语法错误检查模型后，我们的模型相比较 BERT-GED (SRC) 取得了更为显著的提升 (3%-5% 的提升)，并达到了语法错误检查的最好水平。这证明了利用 VERNet 能够从多个语法纠错结果中抽取有价值的语法纠错线索，从而对可能的语法错误起到更好的提示以及辅助作用，进而帮助模型提升其在语法错误检查任务上的效果。采用更强的预训练语言模型 ELECTRA^[101]，VERNet 模型的效果有了进一步提升，证明了我们模型的泛化能力。

表 3.6 不同语法错误检查模型在词级别语法纠错质量评估任务的结果。

模型	FCE test set			CoNLL-2014 ann. 1			CoNLL-2014 ann. 2		
	P	R	F _{0.5}	P	R	F _{0.5}	P	R	F _{0.5}
BERT-GED (HYP)	80.27	40.58	67.14	74.28	34.20	60.17	66.49	27.68	51.93
BERT-GED (JOINT)	76.71	46.94	68.07	71.15	38.30	60.73	64.79	31.52	53.50
BERT-VERNet	81.85	44.27	69.97	76.03	34.02	60.97	71.79	29.04	55.46
ELECTRA-VERNet	80.62	49.16	71.48	74.80	39.26	63.33	72.55	34.42	59.39

3.4.5 语法纠错质量评估效果

在这一部分的实验中，我们进一步评价我们的模型在语法纠错质量评估场景下的效果。我们分别在词级别和句级别的语法纠错质量评估场景下进行模型测试。

首先我们对词级别的语法纠错质量评估效果进行测试，如表3.6所示。两种基线模型 BERT-GED (HYP) 以及 BERT-GED (JOINT) 被比较。其中，对于两个模型 BERT-GED (JOINT) 和 BERT-GED (HYP)，其唯一的区别就是在训练模型的时候是否加入输入句子的语法质量标注标签作为监督信号。我们可以看到 BERT-GED (JOINT) 表现出比 BERT-GED (HYP) 更好的性能，从而证明语法错误检查相关的监督信号能够提升语法纠错质量评估的效果，相类似的现象也在前人的工作中被观察到^[22]。BERT-VERNet 能够进一步提升词级别语法纠错质量评估效果，从而证明来自于语法纠错模型柱搜索解码所提供的多个语法纠错结果可以为估计语法纠错质量提供更多有价值的线索。

我们进一步评价不同语法纠错质量评估模型在句子级别语法纠错质量评估任务上的效果。我们设计了两种场景来评价模型在句级别的语法纠错质量评估任务上的效果：(1) 首先，我们根据不同语法纠错质量评估模型对语法纠错结果的打分来对语法纠错模型通过柱搜索解码所产生的语法纠错结果重新排序，随后对排序结果中排名第一位的语法纠错结果进行语法纠错的相关评价指标的计算；(2) 我们进一步参考已有的语法纠错质量评估的工作^[80]，通过计算质量评估模型对每个输入句子通过语法纠错模型柱搜索解码得到的语法纠错结果列表的打分与真实的语法纠错评价指标得分 ($F_{0.5}$) 之间的皮尔逊相关系数 (Pearson Correlation Coefficient, PCC)^[100] 来评价语法纠错质量评估模型的效果。

我们所采用的基线可以分为两类：基于语言模型和基于语法纠错准确性的语法纠错质量评估模型。前者专注于语法合理性以及流利程度，包括 BERT-LM^[78]，BERT-GQE^[79] 和 BERT-GED (SRC) 三种模型。后者则是通过进一步考虑语法纠错的准确性，以此来避免多余的修改，包括 NQE^[80]，BERT-QE，BERT-GED (HYP) / (JOINT) 三种模型。

如表3.7所示，我们可以发现基于语言模型的语法纠错质量评估模型给出的语法纠错结果的召回率 (Recall) 较高，但是准确度 (Precision) 相对较低，这样的语法纠错质量评估模型提供的语法纠错结果中包含了更多冗余的语法错误修改，这些冗余的语法错误修改建议很可能会误导用户。因此，在语法纠错质量评估任务中，仅仅考虑语法是否符合规范是不够的。与基于语言模型的语法纠错质量评估模型不同，基于语法纠错准确性的质量评估模型可以获得更好的准确度 (Precision) 和更高的 $F_{0.5}$ (我们的主要评价指标)，从而证明了其在语法纠错质量评估任务上的有效性。此外，我们的实验结果发现 BERT-GED (HYP) 优于 BERT-QE，这表明词级别的监督信号能够提供更细粒度的监督信号，从而更好地指导模型区分不同语法纠错结果之间的细微差异。我们的 VERNet 模型能够显著地超过了所有语

表 3.7 不同语法改错质量评估模型在句子级别质量评估效果。

模型	CoNLL-2014 (M ²)			FCE			BEA19			JFLEG
	P	R	F _{0.5}	P	R	F _{0.5}	P	R	F _{0.5}	GLEU
NQE (RR) ^[80]	61.38	33.03	52.39	51.43	30.36	45.16	57.22	46.33	54.65	55.90
NQE (RC) ^[80]	60.09	33.11	51.67	53.97	31.35	47.17	57.87	47.24	55.37	56.91
NQE (CR) ^[80]	62.52	35.24	54.14	51.77	31.46	45.85	57.92	47.43	55.47	56.92
NQE (CC) ^[80]	60.62	35.77	53.23	50.21	32.09	45.11	56.83	49.47	55.19	57.22
BERT-LM ^[78]	52.82	49.59	52.14	36.97	43.42	38.10	46.32	64.05	49.03	59.72
BERT-GQE ^[79]	52.67	50.39	52.19	36.05	43.53	37.33	46.15	64.01	48.88	60.17
BERT-GED (SRC)	52.98	52.07	52.79	37.58	45.81	38.98	47.15	65.09	49.90	60.32
BERT-QE	62.24	38.27	55.31	52.01	36.89	48.07	58.63	54.19	57.69	59.73
BERT-GED (HYP)	68.90	34.35	57.36	57.21	36.03	51.19	68.18	53.85	64.73	60.00
BERT-GED (JOINT)	69.33	36.02	58.51	58.53	37.24	52.53	66.80	55.09	64.07	60.49
BERT-VERNet	68.75	40.26	60.22	58.32	39.99	53.42	66.86	58.60	65.02	61.36
ELECTRA-VERNet	69.97	42.12	61.80	58.77	41.86	54.37	69.09	60.91	67.28	61.61

法改错质量评估基线模型，这个实验结果进一步支持了我们的观点，即充分地利用语法改错模型柱搜索解码所产生的多个语法改错结果，能够提供更多的语法改错线索来对当前的语法改错结果进行质量评估。

我们进一步测试了不同语法改错质量评估模型的打分与真实的语法改错评价指标打分之间的相关性，如表3.8所示。我们可以看到基于语言模型和基于语法改错准确性的质量估计模型在不同数据集，甚至相同数据集的不同标注者所提供的标注结果的皮尔逊相关系数（PCC）上表现都是各不相同的，这个现象说明各个数据集在标注时候也存在一定的偏差。即使数据集的标注规范不尽相同，我们的VERNet模型依旧能够显著地超过全部的语法改错质量评估基线模型，进一步说明VERNet模型能够提供与语法改错结果的真实得分相关性更高的语法改错质量评估结果，从而证明了我们的模型VERNet在语法改错质量评估任务上面的有效性。

3.4.6 通过语法改错质量评估模型提升语法改错效果

本小节探讨了VERNet模型对于提升语法改错效果的有效性。我们通过对来自我们基线语法改错模型的语法改错结果得分以及VERNet所提供的语法改错质量评估分数通过坐标上升方法（Coordinate Ascent, CA）进行加权求和，并将其标记为VERNet[†]模型，进而对基线语法改错模型在柱搜索解码中所提供的全部语法改错结果进行重新排序，排名最高的语法改错结果被保留，进行语法改错相关指标的评测。BERT-fuse (GED)^[61]同样基于我们的基线语法改错模型^[19]并将BERT所提供的对于输入句子的隐状态表示引入到基线语法改错模型中。

表 3.8 语法改错质量评估模型打分与真实语法改错评价指标得分 ($F_{0.5}$) 之间的皮尔逊相关系数。

模型	CoNLL-2014 (M^2)		FCE	JFLEG
	标注 1	标注 2		
NQE (RR) ^[80]	0.2343	0.0662	0.2874	0.0129
NQE (RC) ^[80]	0.2412	0.0552	0.3120	0.0166
NQE (CR) ^[80]	0.2480	0.0912	0.3069	0.0648
NQE (CC) ^[80]	0.2294	0.0839	0.2923	0.0768
BERT-LM ^[78]	0.0347	0.1762	0.0859	0.2685
BERT-GQE ^[79]	0.0256	0.1454	0.1018	0.2905
BERT-GED (SRC)	0.0378	0.2056	0.1271	0.2728
BERT-QE	0.2285	0.1217	0.3384	0.2616
BERT-GED (HYP)	0.3006	0.1679	0.4348	0.2990
BERT-GED (JOINT)	0.2862	0.1628	0.4508	0.3303
BERT-VERNet	0.3102	0.2275	0.4719	0.3698
ELECTRA-VERNet	0.3718	0.2877	0.4812	0.3863

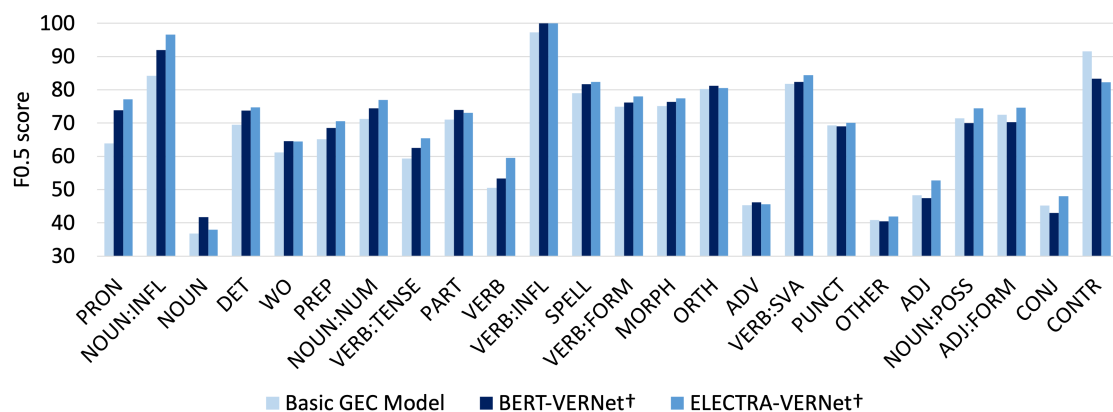
如表3.9所示, 在实验中我们比较了两个基线模型, 即基线语法改错模型^[19]和 BERT-fuse (GED)^[61]。与 BERT-fuse (GED) 相比, BERT-VERNet[†] 在 CoNLL-2014 数据集上取得了几乎相同的效果, 同时在 BEA19 数据集上取得了更大的提升, 这个现象表明使用 BERT-VERNet[†] 为提升语法改错模型性能提供了一种更为有效的方法, 其可以在不更改 Transformer 基础架构的情况下提高语法改错的性能。R2L 模型合并了四个从右到左的基于 Transformer 的语法改错模型, 以进一步提升语法改错的性能。但是, 这些模型使用未发布的数据进行训练, 并且这些模型没有在其开放源代码中提供, 因此这些结果很难重新复现。ELECTRA-VERNet[†] 仅仅融合了一个模型就能够在 BEA19 以及 JFLEG 两个数据集上达到了与之可比较的语法改错效果, 进一步证明了 VERNet 模型的有效性。

我们进一步展示了 VERNet[†] 模型在不同语法错误类型上的表现, 如图3.4所示。其中所有的语法错误类型和它们的缩写都来自于自动化语法改错信息标注工具 ERRANT^[96]。我们绘制了基线语法改错模型和 VERNet[†] 模型在 BEA19 测试数据集的不同语法错误类型上的 $F_{0.5}$ 分数。

VERNet[†] 模型在绝大多数的语法错误类型上都能够提升基线语法改错模型的性能, 尤其是在单词形态等语法错误类型上表现更好, 例如名词错误 (Noun Inflection, NOUN:INFL) 和代词错误 (Pronoun, PRON) 等等。这样的实验结果说明

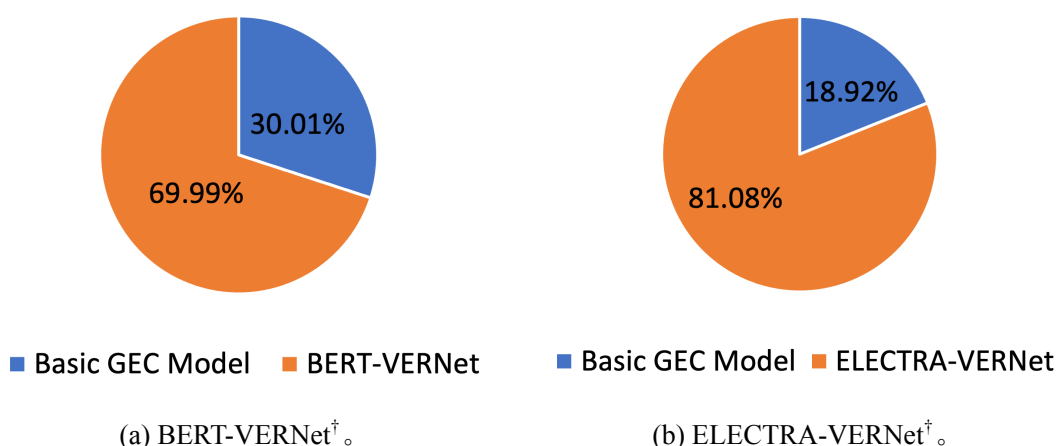
表 3.9 VERNet 模型对语法改错系统的影响。

模型	CoNLL-2014 (M ²)			FCE			BEA19			JFLEG
	P	R	F _{0.5}	P	R	F _{0.5}	P	R	F _{0.5}	GLEU
Basic GEC	68.59	44.87	62.03	55.11	41.61	51.75	66.20	61.48	65.20	61.00
Basic GEC w. R2L	72.4	46.1	65.0	-	-	-	74.7	56.7	70.2	61.4
BERT-fuse (GED)	69.2	45.6	62.6	-	-	-	67.1	60.1	65.6	61.3
BERT-fuse (GED) w. R2L	72.6	46.4	65.2	-	-	-	72.3	61.4	69.8	62.0
BERT-VERNet [†] (Top2)	69.98	43.69	62.47	58.57	41.53	54.13	68.42	60.32	66.63	61.17
BERT-VERNet [†] (Top3)	70.49	43.16	62.57	59.20	41.53	54.55	69.03	60.20	67.06	61.24
BERT-VERNet [†] (Top4)	70.79	42.72	62.56	59.55	41.55	54.80	69.43	60.17	67.36	61.16
BERT-VERNet [†] (Top5)	70.60	42.50	62.36	59.68	41.48	54.86	69.39	60.12	67.32	61.10
ELECTRA-VERNet [†] (Top2)	71.21	44.24	63.47	58.31	41.97	54.09	69.27	61.22	67.50	61.60
ELECTRA-VERNet [†] (Top3)	71.87	44.13	63.84	59.02	41.99	54.59	70.64	61.78	68.67	61.80
ELECTRA-VERNet [†] (Top4)	71.85	43.81	63.69	59.65	42.12	55.07	70.96	62.03	68.98	62.05
ELECTRA-VERNet [†] (Top5)	71.58	43.57	63.43	59.95	42.19	55.29	70.79	61.74	68.77	62.07

图 3.4 VERNet[†] 模型在 BEA19 数据上不同语法错误类型上的表现。

VERNet[†] 模型能够利用从多个语法改错结果中找到的有价值的语法改错线索，以此来更加准确地验证每个词语的语法改错的质量。但是，我们还发现 VERNet[†] 模型在某些语法错误类型，比如缩略语（Contraction, CONTR），上降低了基线语法改错模型的性能。我们认为其原因在于标注者对此种语法错误的标注规范的不一致。对于缩略语（Contraction, CONTR）错误来讲，例如：对于“n’t”和“not”而言，根据语法规则它们都是正确的，但是标注者通常会使用不同的标注标准对其提出不同的语法改错方案。不同的标注规范的差别可能导致了 VERNet[†] 模型在这两个语法错误类型上效果的下降。

如图3.5所示，我们进一步绘制了在 BERT-VERNet[†] 以及 ELECTRA-VERNet[†] 两个模型中不同的柱搜索结果排序特征对于语法改错结果重排序分数的贡献程度，以此来评估语法改错质量评估模型 VERNet 在语法改错结果重排序中的重要性。在

图 3.5 VERNet[†] 模型中不同特征权重分布。

本实验中，我们取每一维特征的权重的绝对值，并计算出其在全部排序特征中的权重占比。我们的实验结果发现 VERNet 在 BERT-VERNet[†] 以及 ELECTRA-VERNet[†] 两个模型中占有最高比例的权重，并且其权重几乎是基线语法纠错模型得分的两倍。这个现象表明，我们的模型 VERNet 能够提供高质量的语法纠错质量评估结果，并在提升语法纠错模型的效果中起到了至关重要的作用。相比较 BERT-VERNet，ELECTRA-VERNet 模型通过其更加准确的语法纠错质量评估结果在最终的重排序分数中占有更多的权重。

表 3.10 融合多种基于 BERT 的语法纠错质量评估模型对语法纠错效果影响。

模型	CoNLL-2014 (M ²)			FCE			BEA19			JFLEG
	P	R	F _{0.5}	P	R	F _{0.5}	P	R	F _{0.5}	GLEU
Basic GEC Model	68.59	44.87	62.03	55.11	41.61	51.75	66.20	61.48	65.20	61.00
BERT-VERNet [†] (Top2)	69.98	43.69	62.47	58.57	41.53	54.13	68.42	60.32	66.63	61.17
BERT-VERNet [†] (Top3)	70.49	43.16	62.57	59.20	41.53	54.55	69.03	60.20	67.06	61.24
BERT-VERNet [†] (Top4)	70.79	42.72	62.56	59.55	41.55	54.80	69.43	60.17	67.36	61.16
BERT-VERNet [†] (Top5)	70.60	42.50	62.36	59.68	41.48	54.86	69.39	60.12	67.32	61.10
BERT-ALL (Top2)	70.85	42.40	62.47	57.43	41.70	53.40	68.97	60.37	67.06	61.64
BERT-ALL (Top3)	72.02	41.32	62.70	58.14	42.10	54.02	69.74	60.71	67.72	62.09
BERT-ALL (Top4)	72.06	40.77	62.47	58.61	42.03	54.33	70.03	60.31	67.85	62.29
BERT-ALL (Top5)	72.23	40.59	62.48	58.62	41.92	54.30	69.96	59.91	67.69	62.38

3.4.7 融合多语法纠错质量评估模型特征对语法纠错效果影响

在这组实验中，我们进一步探究了 VERNet 在基于 BERT 的语法纠错质量评估模型中对语法纠错结果重排序的有效性。我们通过对基线语法纠错模型和不同

的基于 BERT 的语法纠错质量评估模型的评分进行加权和聚合来对基线语法纠错模型柱搜索解码产生的语法纠错结果进行重排序。

在实验中，我们依旧采用坐标上升方法 (Coordinate Ascent, CA) 进行特征整合，并比较两种模型的效果：(1) 仅加上基于 VERNet 的语法纠错质量评估得分 (我们将其表示为 BERT-VERNet[†]) (2) 加上所有基于 BERT 的语法纠错质量评估得分 (我们将其表示为 BERT-ALL)，这里所用到的基于 BERT 的语法纠错质量评估模型包含了 BERT-LM^[78]、BERT-GQE^[79]、参考 NQE^[80] 所实现的 BERT 版本的模型 BERT-QE、BERT-GED (SRC)、BERT-GED (HYP)、BERT-GED (JOINT) 以及 BERT-VERNet。

如表3.10所示，两种模型 BERT-VERNet[†] 和 BERT-ALL 都能够通过重排序的方式进一步提高语法纠错模型在所有语法纠错数据集上的效果，特别是在 FCE 和 BEA19 数据上。这些方法主要着眼于提高语法纠错的准确率 (Precision)，以此来为用户提供更准确的语法错误修正结果。此外，从实验中我们发现，对于仅仅添加 BERT-VERNet 语法纠错质量评估特征的模型 BERT-VERNet[†]，其效果与添加所有基于 BERT 的语法纠错质量评估模型打分的 BERT-ALL 模型的性能相当，进一步证明了 BERT-VERNet 在基于 BERT 的语法纠错质量评估模型中的有效性。BERT-ALL 模型在 CoNLL-2014 数据集以及 JFLEG 数据集上表现更好，证明了通过整合多种语法纠错质量评估模型能够为提升语法纠错效果提供一个行之有效的方法。

Node1 (17.41%) Do one who suffered from this disease keep it a secret of in ##f ##rom their relatives ?
[SEP] Does someone who suffered from this disease keep it a secret from their relatives ? [SEP]
Node2 (24.09%) Do one who suffered from this disease keep it a secret of in ##f ##rom their relatives ?
[SEP] Does anyone who suffers from this disease keep it a secret from their relatives ? [SEP]
Node3 (20.07%) Do one who suffered from this disease keep it a secret of in ##f ##rom their relatives ?
[SEP] Does anyone who suffered from this disease keep it a secret from their relatives ? [SEP]
Node4 (17.67%) Do one who suffered from this disease keep it a secret of in ##f ##rom their relatives ?
[SEP] Does one who suffered from this disease keep it a secret from their relatives ? [SEP]
Node5 (20.77%) Do one who suffered from this disease keep it a secret of in ##f ##rom their relatives ?
[SEP] Does someone who suffers from this disease keep it a secret from their relatives ? [SEP]

图 3.6 VERNet 模型中两种注意力机制权重的可视化结果。

3.4.8 VERNet 模型中注意力机制样例分析

我们从 CoNLL-2014 评测数据中选择一个样例，并可视化节点交互注意力机制和节点选择注意力机制的权重，以研究 VERNet 如何从基线语法纠错模型基于柱搜索解码所产生的多个语法纠错结果中抽取到可以作为语法纠错质量评估的语法纠错证据。

如图3.6所示,我们将输入语句和相应的语法改错结果进行拼接并作为节点(这里句子结尾标识符 “[SEP]” 也参与到注意力权重的计算中)。对 BERT-VERNet 所选择的语法改错结果所在的节点(Node2)我们进行了标注。对语法改错线索进行加权整合的节点注意力机制(Node Selection Attention)我们用深橙色进行标注。对于第二个节点(Node2)中被修改的词语“suffers”,我们展示了节点交互注意力机制(Node Interaction Attention)在不同的语法改错结果中对于词语的注意力权重分布,其中深红色表示更高的注意力权重。

对于给定的输入句子“Do one who suffered from this disease keep it a secret of infrom their relatives?”以及来自基线语法改错模型柱搜索解码所产生的五个语法改错结果,我们首先展示了节点选择注意力机制(Node Selection Attention)对不同语法改错结果的选择作用。对于节点选择注意力机制,其更加关注第二个节点(Node2),根据上下文,我们可以看到第二个节点给出的修改“suffers”相比较其他语法改错结果中的修改“suffered”更加合适于上下文环境,并更加符合语法规则,从而证明了节点选择注意力机制的有效性。

我们进一步展示了面向第二个节点(Node2)中被修改的词语“suffers”的节点交互注意力机制(Node Interaction Attention)对不同的语法改错结果中词语所分配的注意力权重。节点交互注意力机制会准确地从第一、三、四、五个节点中找到能够支持第五个节点(Node5)中将词语“suffered”修改为“suffers”的语法改错线索,“Does”以及“suffers”。二者能够表示当前时态为现在时,以此来提供充分的语法改错证据来支持二个节点(Node2)中将“suffered”改为“suffers”的语法改错方案。

从这个实验中我们可以看到 VERNet 所采用的两种注意力机制,节点交互注意力机制以及节点选择注意力机制,十分直观且有效。这进一步证明了 VERNet 能够很好地建立多个语法改错结果之间的交互,以此来抽取不同语法改错结果中的语法改错线索,进而辅助语法改错质量评估。

3.4.9 语法改错样例分析

如表3.11所示,我们从 CoNLL-2014 数据集中选择了三个语法改错样例,以此来观察我们的 VERNet[†] 模型和基线语法改错模型所产生的语法改错结果的不同,从而评价我们的模型在语法改错任务上的有效性。其中被编辑的部分被标记为红色,被删除的部分同样被标记出来。Golden0 以及 Golden1 表示了 CoNLL-2014 数据集所提供的两个人工标注结果。

对于第一个例子,我们的基线语法改错模型提出了冗余的语法错误修正方案:其将介词“to”添加到了动词“extended”后面。很显然,在给定的输入句子中“vertically”是一个副词,其被用来修饰动词“extended”。我们的 VERNet[†] 模型能够避

表 3.11 VERNet[†] 模型提供的语法纠错结果样例分析。

#1 输入句子	The notion of authority also extended 'vertically' .
Basic GEC Model	The notion of authority also extended to 'vertically' .
BERT-VERNet [†]	The notion of authority also extended 'vertically' .
ELECTRA-VERNet [†]	The notion of authority also extended 'vertically' .
Golden0	The notion of authority also extends 'vertically' .
Golden1	The notion of authority also extended 'vertically' .
#2 输入句子	It is not a heavy job for the family if they have prepare their mental first ...
Basic GEC Model	It is not a heavy job for the family if they have to prepare their minds first ...
BERT-VERNet [†]	It is not a heavy job for the family if they have to prepare their minds first ...
ELECTRA-VERNet [†]	It is not a heavy job for the family if they have prepared their minds first ...
Golden0	It is not a heavy job for the family if they have prepared their mentally first ...
Golden1	It is not a difficult job for the family if they have prepared their minds first ...
#3 输入句子	Many of them are busy in work and separate with their children .
Basic GEC Model	Many of them are busy at work and separate from their children .
BERT-VERNet [†]	Many of them are busy at work and separate from their children .
ELECTRA-VERNet [†]	Many of them are busy at work and separated from their children .
Golden0	Many of them are busy in working and separated with their children .
Golden1	Many of them are busy in work and separated from their children .

免产生此类冗余的语法错误修正方案，从而提升了语法纠错的准确性并为用户提供更加精准的语法纠错结果。

第二、三个样例进一步展示了 ELECTRA-VERNet[†] 的良好性能。对于第二个例子，基线语法纠错模型在动词“prepare”之前插入介词“to”，这样的修改虽然能够满足一定的语法规则，即“have”不能直接加动词原形形式。但是通过结合上下文语境，我们发现这样的语法纠错修改并不符合上下文语义。相反，ELECTRA-VERNet[†] 可以找到更加合适的语法纠错方案，即将“prepare”更正为“prepared”。从上述现象中我们可以发现，虽然基线语法纠错模型能够提供更好的语法纠错结果，但是该结果却在柱搜索解码过程中排在了较后的位置上，从而限制了语法纠错模型的效果。我们的模型 ELECTRA-VERNet[†] 能够通过语法纠错质量评估以及语法纠错结果重排序的方法提供一个更优的语法纠错结果。

对于第三个例子，ELECTRA-VERNet[†] 可以将动词“separate”正确地修正为其过去时形式“separated”。这个现象说明了，基于判别器训练方式的预训练语言模型 ELECTRA^[101] 能够更好地理解输入句子的语法以及语义信息，并通过找到距离更远的上下文信息（例如，词语“are”），来进一步判断“separated”在当前语境下的语

法合理性。利用更强大的预训练语言模型（例如 ELECTRA）可以进一步提升语法错误检查以及语法改错质量评估任务上的性能，从而进一步提升语法改错的效果。

3.5 本章小结

在本章中，我们提出了一种基于多语法改错结果的语法错误检查以及语法改错质量评估模型 VERNet。VERNet 通过一个推理图来建立多个语法改错结果之间的交互，然后设计了两种注意力机制，节点选择注意力机制（Node Selection Attention）和节点交互注意力机制（Node Interaction Attention），以此来提取并整合语法改错线索。两种注意力机制分别从多个语法改错结果中抽取和汇总语法改错证据，以验证当前的语法改错结果的质量。我们在四个语法改错数据集上的实验表明，我们的模型 VERNet 达到了当前最好的语法错误检查和语法改错质量评估的效果，并能够显著提升当前表现最好的基于生成的语法改错系统的性能。在未来，我们将进一步研究 VERNet 在其他应用中的有效性，例如论文评分。

第4章 融合多事实证据的细粒度联合推理的事实验证

在前面的章节中，本文主要对怎样充分利用语法敏感的语言模型所提供的语法纠错线索来提升在语法层面的英文文本校对能力的问题进行讨论。除了语法层面的校对外，事实验证在英文文本校对中也发挥其重要作用。事实验证需要借助于信息检索技术从大规模知识库中抽取与当前待验证文本有关的证据，从而进一步结合检索出的多条证据推理出当前待验证文本是否含有事实性错误^[103]。然而，目前大多数关于事实验证方面的研究，都主要集中在如何能够更好地进行多证据联合推理，却忽略了在检索出的每条证据中只有部分语义信息可以验证当前文本的事实正确性。因此，在本章^①中，我们提出了基于核函数的图注意神经网络，它利用基于核函数的注意力机制进行更细粒度的多证据联合推理。

具体来说，给定一个待验证文本和一组检索出的相关证据，我们首先建立推理图来联合多证据进行推理，从而验证出待验证文本是否有事实性错误。我们将基于核函数的向量匹配机制引入到图注意网络中，进一步设计了两种注意力机制。第一种注意力机制（Edge-Kernel）能够在推理图中进行细粒度证据传播，从而更好地进行多证据联合推理。第二种注意力机制（Node-Kernel）能够更好地度量证据节点的重要性。通过两种注意力机制，我们的模型能够更准确地进行验证事实，提升模型的推理能力，并且显著优于现有的事实验证模型。我们的分析表明，与传统的基于点积的注意力机制相比，基于核函数的注意力机制能够更集中地关注到推理图中与待验证文本相关的证据句子中有意义的推理线索，证明了我们的模型在事实验证任务上的有效性。

4.1 问题描述

在英文文本自动校对中，事实检查^[103]作为其中重要一环^[6]，保证了出版刊物的正确性以及可信性。正如英文谚语所说“*If your mother tells you she loves you, check it out.*”，在写作过程中，我们对所记忆的或从别处获得的信息做出过强的假设，过分相信我们的信息来源。这暗示了我们在校对文章或者写作时候经常犯的错误，当我们第一次从某个地方或者从我们不完善的记忆中获得某个知识的时候，我们经常会重复这个知识，直到将这个知识作为“常识”。因此，我们迫切地需要自动地进行事实检查，从而验证文本内容的完整性以及正确性。

^① 本章主要工作以“*Fine-grained Fact Verification with Kernel Graph Attention Network*”为题发表在2020年的国际学术会议 *The Annual Meeting of the Association for Computational Linguistics (ACL 2020)* 上。

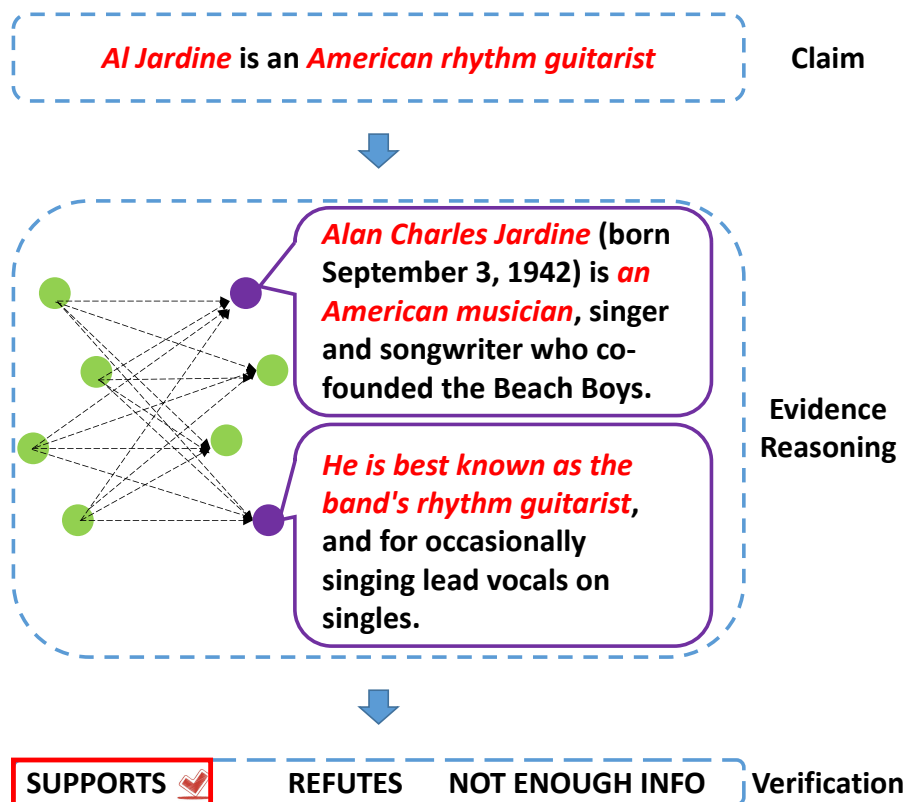


图 4.1 英文文本事实验证的一个例子。

由于存在事实错误的句子在语法或者语义上往往是正确的，因此，我们需要利用外部的可信赖知识库对文本进行事实层面的验证。最近的研究将此问题表述为事实验证任务^[103]，该任务旨在利用可信赖语料库（例如：维基百科^①）来自动验证输入句子的合理性以及完整性。

例如，如图4.1中所示，对于待验证文本“艾尔·贾丁是美国节奏吉他手”，事实验证系统可以首先从可信赖语料库中检索出相关的证据语句，通过整合这些证据语句对待验证文本进行事实验证^[27,29,104-105]。在这里，检索出来的两条证据“艾伦·查尔斯·贾丁（Alan Charles Jardine）（1942年9月3日出生）是美国音乐家，歌手和词曲作者，是海滩男孩（Beach Boys）的共同创始人。”以及“他最出名的职业是作为乐队的节奏吉他手，并在诸如《Help Me, Rhonda》（1965），《Then I Kissed Her》（1965）和《Come Go With Me》等单曲中担任主唱。（1978）。”分别支持了待验证文本中的“艾尔·贾丁是美国音乐家”以及“艾尔·贾丁是节奏吉他手”两个方面的语义表述，进而验证了待验证文本事实正确性以及完整性。待验证文本可以被划分为三个类别，被支持、被拒绝以及没有足够的证据验证文本是否符合事实。然而，在检索出的两条证据中，每一条证据中只有一部分文本与待验

① <http://www.wikipedia.org>

证文本有关，因此，需要进行细粒度推理，从而找到每条证据中真正能够验证待验证文本的事实性线索。

近些年，一些研究者对事实验证任务进行了研究，并利用自然语言推理（Natural Language Inference, NLI）相关技术^[93,106-110]来完成对于文本的事实验证工作。随着预训练语言模型的发展，诸如 BERT, RoBERTa 等预训练语言模型^[15,50,93]也广泛地应用在自然语言处理任务中，显著地提升了自然语言推理任务的效果，极大地推动了自然语言推理方向的发展。受益于预训练语言模型对文本的良好理解能力以及推理能力，事实验证的相关工作也逐渐使用预训练语言模型来提升其准确性^[27-28]。

不同于自然语言推理任务，事实验证需要系统自行查找与待验证文本在事实层面相关的证据语句，而且通常会需要通过多个证据才能够验证出待验证文本的事实正确性^[28,103]。因此，很多事实验证的工作通过直接拼接多条证据^[28]或利用图神经网络模型^[27]来充分地利用检索出来的多条相关证据，从而联合多条证据对当前待验证文本进行事实验证。然而，在事实验证领域还存在着两个重要的挑战。

(1) 由于事实验证参考了开放域问答的相关工作^[25]，因此，现有的事实验证有关的工作也是应用了三步的流水线模型，该模型包含了文档检索、句子检索和事实验证三个步骤。通过文档级别以及句子级别的搜索，我们进一步得到了与待验证文本相关的证据。由于这部分事实证据的获取往往是由信息检索模型来完成的，因此，在该阶段不可避免地引入额外噪音，此种噪音往往会影响事实验证的结果^[111]。事实验证模型希望能够从检索到的相关证据集合中找到真正与待验证文本相关的证据句子，从而更加准确地验证其事实正确性。

(2) 出现事实错误的文本往往保证了语法和语义上的合法性。比如，在一句话中，很可能是一个数字或者是一个名词等比较细微的错误，导致待验证文本往往与相关的事实证据比较相似。因此，在事实验证任务中，模型往往需要识别出待验证文本与相关证据句子之间微小的区别，从而对检索出的多条证据进行细粒度的联合文本推理，进一步提升事实验证的准确性。

为了解决上述两个问题，我们提出了一种基于核函数的图注意力网络（Kernel Graph Attention Network, KGAT）模型来对待验证文本进行事实验证。受到了信息检索领域相关模型的启发^[112-113]，该模型设计了具有不同均值的核函数，以此来表示不同的语义相似度匹配等级。进一步该模型将词语的相似度特征分布到不同的核函数上面，以此来计算两个文本在不同相似度等级上的词语级别的相似度分布特征，以此来计算词语的重要程度。该模型通过对不同的相似度等级建模赋予了基于图神经网络的事实验证模型更细粒度的证据选择能力和推理能力。具体来

说，对于给定检索到的证据片段，我们首先构造一个证据图，将待验证文本和证据文本进行拼接，作为图的节点并且完全连通全部的节点，形成推理图。然后利用两组基于核函数的注意力机制，进行证据的选择和多证据联合推理。我们的模型与基线模型相比，能够在事实验证任务上取得了显著的性能提升。

总结来说，我们的工作有以下三个贡献：

(1) 我们基于核函数设计了 **Edge-Kernel** 注意力机制，其能够有选择地总结每条证据中的线索从而获得更细粒度的证据表示，并通过多层图注意力机制^[27] 在相邻节点之间传播事实验证线索，进一步实现了多证据的联合推理。

(2) 此外，我们设计了 **Node-Kernel** 注意力机制，其通过基于核函数的匹配机制更好地计算待验证文本和事实证据句子的匹配特征，从而能够更加准确地进行证据选择。基于 **Node-Kernel** 的注意力机制能够整合每条证据中的事实验证信号，从而更准确地选择出相关证据进行事实验证。

(3) 在大规模事实验证基准数据集 **FEVER**^[103] 的实验中，**KGAT** 模型显著优于以往基于预训练语言模型 **BERT** 和图形神经网络 (**Graph Neural Network, GNN**)^[114-116] 的方法^[27]，从而验证了模型的有效性。我们的实验证明了，尤其是在需要多证据推理的事实验证场景上，我们的模型的推理效果有了更加明显的提升。我们设计的基于核函数的注意力机制能够提供更集中、更加有意义的注意力模式，这种高效的注意力模式是我们模型的有效性的主要来源。

4.2 相关工作

FEVER 共享任务^[103] 旨在开发自动的事实验证系统，其通过从维基百科中抽取证据来检查待验证文本的真实性和完整性。最近发布的 **FEVER** 共享任务 1.0 版本是以竞赛的形式发布在 **Codalab**^① 上，并以黑盒测试的形式进行算法的评价。**FEVER** 共享任务引起了自然语言处理社区的广泛关注。

现有的工作通常利用广泛使用的事实验证数据集，**FEVER** 数据集^[103]，并且仿照开放域问答的相关工作^[25] 沿用了流水线系统对文本进行事实验证，该系统包含了三步：文档检索，句子检索和事实验证。当前绝大多数的工作都集中在事实验证这一步骤上面。

事实验证这一步的工作主要集中在如何联合多条证据进行联合推理。其相关工作主要包含以下几种。一部分工作通过将多个证据进行拼接^[28,104]，进一步利用自然语言推理模型对待验证文本进行事实验证。另外一些工作分别使用每一条证据对待验证文本进行事实验证，进一步地将每条证据的事实验证标签汇总，从而

① <https://competitions.codalab.org/competitions/18814>

实现联合多个证据的事实验证^[29,105,117]。TwoWingOS^[118]进一步结合了事实验证流水线中的句子检索部分所得到的证据选择概率，从而能够进一步地对无关证据进行过滤。

近期，Zhou 等人提出了基于图注意力网络的事实验证模型 GEAR^[27]，该模型将事实验证任务形式化为基于图的多证据推理问题，并设计了两种注意力机制进行证据的传播以及整合。它使用图模型^[114-116]结合相关证据进行推理并汇总证据信息。Zhong 等人^[30]进一步使用 XLNet^[51]对输入文本和证据进行编码，并通过语义分析 (Semantic Parsing) 建立了语义级推理图以进行细粒度推理，从而取得更好的事实验证效果。这些基于图模型的事实验证工作都是通过将每个证据抽象为一个节点，通过图神经网络建立起节点之间的交互，从而能够联合多个证据进行推理，进一步提升事实验证的准确性。

许多事实验证系统都利用自然语言推理 (Natural Language Inference, NLI) 技术^[93,106-110]来对输入文本进行编码从而实现事实验证模型。自然语言推理任务旨在将一对前提和假设之间的关系归纳为包含、矛盾或中立三类。在事实验证相关的工作中使用最广泛的自然语言推理模型之一是增强序列推理模型 (Enhanced Sequential Inference Model, ESIM)^[106]，该模型采用长短期记忆循环神经网络 (Long Short Term Memory, LSTM)^[14]对前提和假设进行编码，并且采用不同的机制整合局部推断和全局推断。随着预训练语言模型的发展^[15,93,109]，基于 Transformer^[17]的预训练模型 BERT 在许多 NLP 任务中发挥了重要作用。BERT 已经广泛地应用在了事实验证领域，从而提升事实验证的效果^[15,27-28,110]。

近些年在信息检索领域的相关工作，尤其是基于交互的信息检索模型 (Interaction based Neural Ranking Models)^[119]，通过建立问题和文档的交互，以此抽取出问题和文档之间的词语匹配特征，从而提升信息检索效果^[112-113,119-121]。对于文本匹配进行建模的有效方法之一是利用基于核函数的匹配特征计算排序分数^[112-113]。具体来说，基于核函数的匹配特征首先基于问题和文档间的向量化嵌入式表示建立二者之间的单词或短语级别的交互，随后利用不同的核函数抽取出问题和文档的词语或短语的不同相似程度的局部匹配特征。基于核函数的匹配特征抽取模式，在各种信息检索数据集中表现出良好的性能^[122]。最近的研究还表明，可以将基于核函数的匹配特征提取机制与预训练模型 BERT 进一步集成，从而能够更好地对问题和文档之间的相关性进行建模^[123]。

4.3 模型框架

在这一节中，我们将详细介绍基于核函数的图注意力网络（Kernel Graph Attention Network, KGAT）在事实验证中的应用。

针对于待验证文本 c ，KGAT 模型首先使用检索到的相关证据语句构造推理图，然后基于推理图来预测待验证文本 c 的事实验证标签 y 。如图4.2所示，KGAT 模型包含了两个部分，分别为：基于 Edge Kernel 的证据信息传播模块和基于 Node Kernel 的证据选择模块。

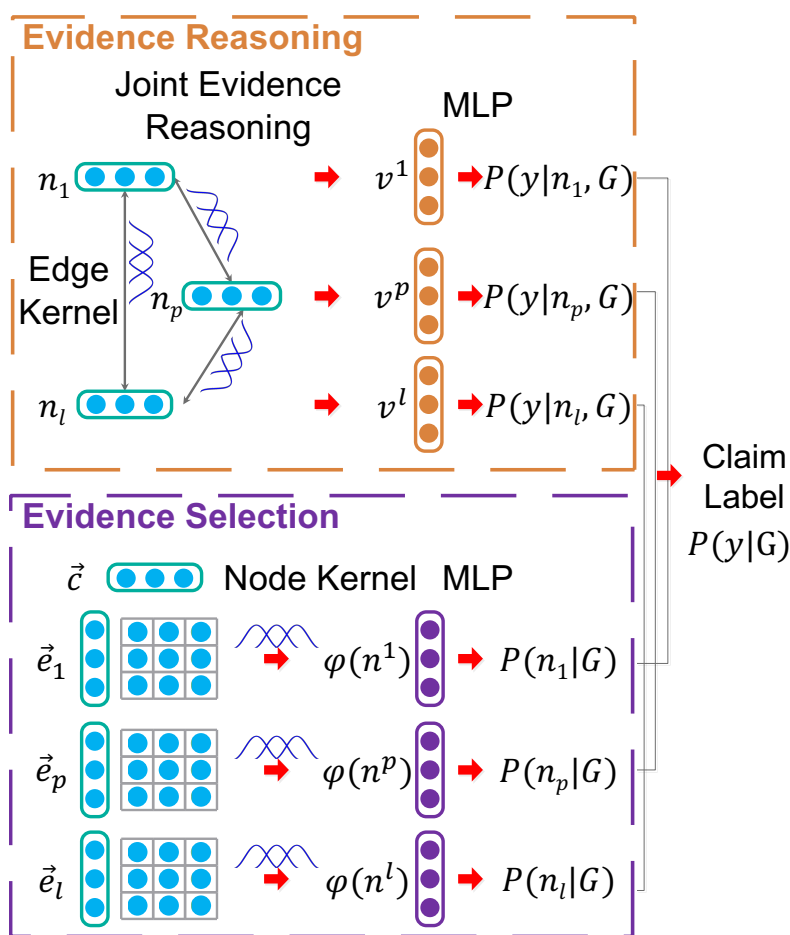


图 4.2 KGAT 模型框架图。

4.3.1 基于推理图的多证据联合推理

类似于 Zhou 等人的提出来的基于图网络的事实验证模型 GEAR^[27]，对于检索出来的与待验证文本 c 相关的 l 条证据 $D = \{e^1, \dots, e^p, \dots, e^l\}$ ，KGAT 首先将待验证文本 c 与每个检索到的证据 e^p 进行拼接，并将其作为一个节点 n^p 。因此，我们可以得到包含 l 个节点的节点集合 $N = \{n^1, \dots, n^p, \dots, n^l\}$ ，随后我们将所有节点进行全连接，从而构造出针对于待验证文本 c 的推理图 G 。

KGAT 统一定义了在多证据联合推理和单证据推理场景下的事实验证框架, 并通过计算概率 $P(y|c, D)$ 来基于检索出的相关证据集合 D , 预测待验证文本 c 的事实验证标签 y 。不同于 Zhou 等人提出的 GEAR^[27] 模型, 我们遵从了在图神经网络中, 基于图的标签预测方法的标准设置^[114], 并将事实标签预测概率 $P(y|c, D)$ 依据贝叶斯公式进行分解, 采用基于推理图的符号定义, 进一步重新将事实验证标签预测概率的计算分解为两个部分:

- 1) 基于整个推理图 G 的单一节点的标签预测概率 $P(y|n^p, G)$;
- 2) 以及针对每个节点置信度的节点选择概率 $P(n^p|G)$ 。

随后我们可以进一步地对事实验证进行形式化定义:

$$P(y|c, D) = \sum_{p=1}^l P(y|c, e^p, D)P(e^p|c, D), \quad (4-1)$$

或者基于推理图标记的事实验证标签预测:

$$P(y|G) = \sum_{p=1}^l P(y|n^p, G)P(n^p|G). \quad (4-2)$$

其中, 事实验证标签预测的联合推理概率 $P(y|n^p, G)$ 代表使用多个证据联合对待验证文本 c 进行事实验证时候的事实验证标签预测的计算过程。我们设计的读出模块^[124] 计算每个节点 n^p 的被选择概率 $P(n^p|G)$, 并根据概率 $P(n^p|G)$ 融合每一个节点的预测标签, 进一步预测待验证文本的事实验证标签 y 。

本节的其余部分介绍了对于推理图 G 中每个节点 n^p 表示方法, 通过 Edge Kernel 来预测每个节点的事实验证标签 $P(y|n^p, G)$ 的计算过程, 以及通过 Node Kernel 对推理图 G 中的节点进行选择时候的概率 $P(n^p|G)$ 的计算过程。

4.3.2 推理图节点表示初始化

此部分详细介绍了如何得到每个节点 n_p 的表示。我们首先通过将待验证文本 c 、每条证据所对应的文档标题 (这里是每一个维基百科文章的标题) 和相关性证据进行拼接, 进一步作为输入并使用预训练语言模型 BERT^[15] 来初始化每个节点的表示。具体而言, 在每个节点 n_p 中, 待验证文本 c 和相应的证据文本 e^p 分别包含了 m 个词语 (其中包含了 BERT 的句尾标识符 “[SEP]”) 和 n 个词语 (其中包含了证据所对应的维基百科页面的标题和句尾标识符 “[SEP]”)。使用 BERT 编码器, 我们得到给定节点 n^p 的词语级别的隐状态表示 H^p :

$$H^p = \text{BERT}(n^p). \quad (4-3)$$

其中, 每一个节点 n^p 中的第一个词语 “[CLS]” 的表示被作为节点的初始化表

示：

$$z^p = H_0^p. \quad (4-4)$$

除了第一个词语之外的词语的表示 $H_{1:m+n}^p$ 也被用来表示当前的节点 n^p 。其中， $H_{1:m}^p$ 代表了待验证文本 c 中的词语表示， $H_{m+1:m+n}^p$ 则是代表了证据文本 e^p 中的词语表示。

4.3.3 基于 Edge Kernel 的证据信息传播

KGAT 中的证据传播和每个节点的事实验证标签预测是通过 Edge Kernel 来实现的，Edge Kernel 利用基于核函数的注意机制沿着在推理图 G 中的节点之间的连边进行节点之间推理证据的传播。

具体来说，KGAT 使用基于核函数的注意机制计算每一个节点 n^p 的融合多证据信息的节点表示 v^p ，并使用该节点表示来计算当前节点对待验证文本 c 的事实验证标签 y 的预测概率：

$$\begin{aligned} v^p &= \text{Edge-Kernel}(n^p, G), \\ P(y|n^p, G) &= \text{softmax}_y(\text{Linear}(v^p)). \end{aligned} \quad (4-5)$$

KGAT 的 Edge Kernel 采用分层注意机制在节点间传递推理信息。它使用基于词级别的注意力机制生成细粒度的节点表示，并使用句子级别的注意力机制沿着在推理图 G 中的节点之间的连边传播推理信息。

词级别的注意力机制。对于推理图 G 中的每一个节点 n^p ，词级别的注意力机制使用了基于核函数的注意力机制，用于控制每个节点向其他节点传播的推理信息。基于核函数的注意力机制能够计算出在推理图 G 中与节点 n^p 相邻接的节点 n^q 的细粒度节点表示 $z^{q \rightarrow p}$ 。

为了计算出节点 n^p 对相邻接的节点 n^q 中词级别的注意力权重 $\alpha^{q \rightarrow p}$ ，对于节点 n^p 的相邻接的节点 n^q 中的第 i 个词语，我们首先在节点 n^p 和节点 n^q 之间建立一个词级别的节点之间的交互矩阵 $M^{q \rightarrow p}$ 。对于交互矩阵 $M^{q \rightarrow p}$ 中的每一个元素 $M_{ij}^{q \rightarrow p}$ 来说，其都是通过对应的节点中词语的隐状态表示 H_i^q 和 H_j^p 的余弦相似度计算出来的：

$$M_{ij}^{q \rightarrow p} = \cos(H_i^q, H_j^p). \quad (4-6)$$

然后，我们使用 K 个核函数来提取两个节点的交互矩阵 $M^{q \rightarrow p}$ 中的文本匹配特征 $\vec{K}(M_i^{q \rightarrow p})$ ^[32,112-113,123]：

$$\vec{K}(M_i^{q \rightarrow p}) = \{K_1(M_i^{q \rightarrow p}), \dots, K_K(M_i^{q \rightarrow p})\}. \quad (4-7)$$

每个核函数 K_k 使用高斯核函数 (Gaussian Kernel) 来提取节点 n^p 和节点 n^q 之间的文本匹配特征, 该匹配特征是一种词级别的相似度分布统计, 每一个核函数表示词语之间不同程度的语义匹配:

$$K_k(M_i^{q \rightarrow p}) = \log \sum_j \exp\left(-\frac{M_{ij}^{q \rightarrow p} - \mu_k}{2\delta_k^2}\right), \quad (4-8)$$

这里 μ_k 和 δ_k 分别代表了第 k 个核函数的均值和方差。基于核函数的注意力机制能够通过不同的核函数捕捉词语之间的不同程度的语义相似度^[112]。

然后, 我们使用线性层 (Linear Layer) 计算节点 n^p 中第 i 个词语 n_i^p 对相邻接的节点 n^q 中的词语 n_j^q 的词级别的注意力权重 $\alpha_i^{q \rightarrow p}$:

$$\alpha_i^{q \rightarrow p} = \text{softmax}_i(\text{Linear}(\vec{K}(M_i^{q \rightarrow p}))). \quad (4-9)$$

词级别的注意力权重 $\alpha_i^{q \rightarrow p}$ 被用来整合节点 n^q 的词语表示, 进一步得到节点 n^q 的细粒度节点表示 ($\hat{z}^{q \rightarrow p}$):

$$\hat{z}^{q \rightarrow p} = \sum_{i=1}^{m+n} \alpha_i^{q \rightarrow p} \cdot H_i^q, \quad (4-10)$$

我们在这里将节点 n^q 的内容进行细粒度编码, 从而控制从节点 n^q 到节点 n^p 的推理证据的传播。

句级别注意力机制。句级别注意力机制进一步将在推理图 G 中与节点 n^p 相邻节点 n^q 的细粒度节点表示进一步汇总并整合, 从而形成对于节点 n^p 以及其相连节点的联合推理表示 v^p 。与以前的工作^[27] 相同, 我们也是通过一个图注意机制^[114] 来完成节点的聚合工作。

句级别注意力机制首先计算了节点 n^p 对其相邻接节点 n^q 的注意力权重 $\beta^{q \rightarrow p}$:

$$\beta^{q \rightarrow p} = \text{softmax}_q(\text{MLP}(z^p \circ \hat{z}^{q \rightarrow p})), \quad (4-11)$$

在这里 \circ 代表了拼接操作, z^p 是节点 n^p 的初始化表示。

然后, 通过将邻居节点表示 $\hat{z}^{q \rightarrow p}$ 使用注意力权重 $\beta^{q \rightarrow p}$ 进行加权求和即可以得到节点 n^q 的融合多证据信息的节点表示 v^p :

$$v^p = \left(\sum_{q=1}^l \beta^{q \rightarrow p} \cdot \hat{z}^{q \rightarrow p}\right) \circ z^p. \quad (4-12)$$

这里, 我们利用了邻居节点的细粒度节点表示 ($\hat{z}^{q \rightarrow p}$) 对当前节点 n^p 的节点表示进行更新, 并得到更新后的多证据节点表示 v^p 。该表示是由基于核函数的词级别的注意力机制计算 (如公式4-9所示), 随后通过一个句子级别的注意力机制进行多证据的整合 (如公式4-11所示)。

基于节点的事实标签预测。对于节点 n^p , 更新后的多证据节点表示 v^p 被用来

计算节点 n^p 的事实验证标签 y 的概率 $P(y|n^p)$:

$$P(y|n^p, G) = \text{softmax}_y(\text{Linear}(v^p)). \quad (4-13)$$

由于每个节点 n^p 的表示 v^p 是通过汇总在推理图 G 中与之相邻的节点中的推理信息，随后进一步来更新节点 n^p 的表示所获得的。因此，每个节点 n^p 的事实验证标签概率的预测也取决于与其相关联的节点所提供的推理信息。综上所述，Edge Kernel 能够实现结合多证据对待验证文本 c 进行联合推理验证。

4.3.4 基于 Node Kernel 的证据选择

为了整合每一个节点 n^p 对事实验证标签 y 的预测结果，我们进一步设计了基于核函数的注意力机制 Node Kernel 来对每一个节点的预测结果进行“读出”，并进一步整合。在这里 Node Kernel 能够对每一个节点的重要程度进行衡量。

Node Kernel 注意力机制首先使用与 Edge Kernel 相同的一组核函数来计算每一个节点 n^p 中的待验证文本 c 和相关证据 e^p 的匹配特征表示 $\phi(n^p)$ ，并进一步将其作为证据选择时候的节点表示：

$$\phi(n^p) = \text{Node-Kernel}(n^p). \quad (4-14)$$

类似于 Edge Kernel，我们首先建立了节点 n^p 中的待验证文本 c 和相关证据 e^p 之间的交互矩阵 $M^{c \rightarrow e^p}$ ，其中每一个元素都是使用待验证文本 c 和相关证据 e^p 相对应的词级别的表示 $H_{1:m}^p$ 和 $H_{m+1:m+n}^p$ 的余弦相似度计算得到。基于待验证文本 c 和相关证据 e^p 的交互矩阵 $M^{c \rightarrow e^p}$ ，我们进一步使用核函数抽取节点 n^p 中的待验证文本 c 和相关证据 e^p 之间的相关性特征表示 $\phi(n^p)$ ，并将其作为节点的选择概率计算的表示：

$$\phi(n^p) = \frac{1}{m} \cdot \sum_{i=1}^m \vec{K}(M_i^{c \rightarrow e^p}). \quad (4-15)$$

节点 n^p 中的待验证文本 c 和相关证据 e^p 之间的匹配特征 $\phi(n^p)$ 被用来计算出其在整个推理图 G 被选择的概率 $P(n^p|G)$:

$$P(n^p|G) = \text{softmax}_p(\text{Linear}(\phi(n^p))). \quad (4-16)$$

KGAT 采用了基于核函数的多相似度匹配级别的相似度分布特征计算节点 n^p 中的待验证文本 c 和相关证据 e^p 的匹配特征^[112]，进一步计算出两者的相关性，最后根据每个节点的重要程度来对每个节点 n^p 的预测结果进行加权整合，从而计算

出待验证文本 c 的事实验证标签 y 的概率 $P(y|G)$:

$$P(y|G) = \sum_{p=1}^l P(y|n^p, G)P(n^p|G). \quad (4-17)$$

整个模型被端到端训练，通过最小化交叉熵损失函数（Cross Entropy Loss）进行参数优化：

$$L = \text{CrossEntropy}(y^*, P(y|G)), \quad (4-18)$$

在这里 y^* 代表了对于待验证文本 c ，其真实的事实验证标签，这些标签都是来自于我们人工标注的事实验证数据集。

4.4 实验

这一节中，我们将介绍在事实验证任务上实验所用到的数据集，实验设置，基线模型以及最终的实验结果。

4.4.1 数据集

在实验中，我们使用了大规模的通用事实验证数据集 FEVER^[103]。FEVER 数据集包括了 185,455 个人工标注的待验证句子以及 2017 年 6 月维基百科中的 5,416,537 个页面文档。数据集的划分与 FEVER 共享任务^[24] 的划分保持相同，如表4.1中所示。

FEVER 数据的标注分为两个阶段，待验证文本生成和事实标签标注。总共有 50 个标注者参与了标注过程。在第一阶段，标注者从维基百科中随机选择句子进行改写，从而形成待验证文本。待验证文本的生成阶段的目标是创建既有一定事实验证难度的文本（即与改写前的句子不要过分相似），同时也不是太复杂的待验证文本。为此，在每个句子中加入了超链接，以便标注者以可控的方式结合外部知识，来对文本进行改写。除对句子进行改写外，标注者还通过，例如：释义，添加否定词等方式，来创建句子变体。第二个阶段则是待验证文本的标注阶段，这个阶段要求标注者将每个句子标注为支持（SUPPORTED），拒绝（REFUTED）或信息不足（NOT ENOUGH INFO, NEI）三种类别。对于支持和拒绝标签，标注者还提供了用作支持或反驳待验证文本真实性的维基百科中的相关证据句子。对于被标记为信息不足的样例，数据集仅仅提供了标注标签，因为标注者无法从维基百科中找到能够支持或反驳待验证文本的证据。最后，为了提高标注数据集的质量，作者进行了如下操作，随机选择 4% 的数据对标注结果的相关性采用 Fleiss 评分^[125] 进行评价，以及人工验证构建数据集中 227 个样本的标注质量。

FEVER 共享任务的官方评估指标^①被用来评价模型在事实验证任务上的效果。其中包括两个评价指标,第一个为事实验证标签的预测准确度(Label Accuracy),第二个则为 FEVER 分数。事实验证标签预测准确度是一种通用评估指标,它在不考虑检索到的证据的情况下,计算每一个待验证文本进行事实验证标签分类的准确率。FEVER 评分进一步地考虑了证据获取阶段是否检索出了能够完整地验证待验证文本的事实正确性的相关证据,因此,FEVER 评分能够更好地反映事实验证模型的推理能力,并作为该任务的主要评价指标。

此外,我们还设计了 Golden FEVER (GFEVER) 分数来评估模型的效果。和 FEVER 分数类似,GFEVER 也期望能够准确地评价模型的推理能力。但是与 FEVER 评分不同的是,在测试模型推理能力的时候,我们为事实验证系统提供能够验证出待验证文本准确性的证据,以此来消除证据获取部分对于事实验证模型的影响。Precision, Recall 和 F1 也用来评价证据获取部分效果,我们利用 FEVER 数据集所提供的相关证据标签来评价证据检索模型效果,并评价其是否有能力提供足够的证据来对待验证文本进行事实验证。

表 4.1 事实验证数据集 FEVER 的训练集、验证集和测试集划分。

数据集划分	# 支持	# 拒绝	# 信息不足
训练集	80,035	29,775	35,639
验证集	6,666	6,666	6,666
测试集	6,666	6,666	6,666

4.4.2 基线模型

我们在实验中所采用的基线模型包括了 FEVER 共享任务 (1.0 版本) 比赛期间的排名前三的事实验证模型和基于 BERT 的事实验证模型。其中 GEAR 模型^[27]为我们的主要基线模型。

Athene^[29] 利用自然语言推理模型 ESIM 获取每个待验证文本和证据句子对的表示,然后利用最大池化 (Max Pooling) 和平均池化 (Mean Pooling) 来整合每条证据的事实验证标签。

UNC NLP^[104] 基于自然语言推理模型 ESIM,提出了神经语义匹配模型 (Neural Semantic Matching Network, NSMN)。该模型通过融合诸如 WordNet 等来源的不同语义知识来提升事实验证的准确性,并达到在 FEVER 共享任务 (1.0 版本) 比赛期间的第一名的成绩。

UCL MRG^[105] 首先利用卷积神经网络 (Convolutional Neural Network, CNN)

^① <https://github.com/sheffieldnlp/fever-scorer>

对待验证文本和证据进行编码。然后对每一条证据分别生成事实验证标签。最后，通过一些简单的规则或学习的权重来汇总所有预测标签。

BERT Pair^[27] 将待验证文本和每一条检索到的证据进行拼接，并输入到 BERT 模型中，以此来预测每个待验证文本和证据文本对的事实验证标签。具体而言，在训练阶段，BERT Pair 模型采用标注数据对模型进行训练。在测试阶段，BERT Pair 模型利用所有检索到的证据分别对待验证文本进行事实验证，并得到事实验证标签。由于不同的证据对当前验证文本可能有不一致的预测标签，因此，BERT Pair 模型使用了一个聚合器来整合所有证据的事实性预测标签，进一步获得待验证文本的事实验证标签。

BERT Concat^[27] 直接将待验证文本和全部检索证据进行拼接，并输入到 BERT 模型中，利用 BERT 的 “[CLS]” 隐状态来预测待验证文本的事实验证标签。在这里 BERT Concat^[27] 基于 ESIM 模型实现句子检索模型。

BERT^[28] 和 BERT Concat^[27] 类似，唯一不同之处在于 BERT^[28] 采用了基于 BERT 的句子检索模型。

SR-MRS^[126] 利用 BERT 模型实现了在句子检索模块以及事实验证模块中对文本的编码。该模型提出了一个简单而有效的事实验证系统，其将文档和句子层次上的分层语义检索信息引入到下游事实验证模块中。

GEAR^[27] 采用 BERT 模型对待验证文本和检索到的证据分别进行编码，利用 BERT 模型所提供的 “[CLS]” 隐状态来表示待验证文本和每一条被检索到的证据。该模型进一步利用多层图注意力神经网络通过联合多个检索到的证据对待验证文本的事实验证标签进行联合推理。在证据整合阶段，GEAR 模型则采用了基于注意力机制的证据信息整合机制。

GAT 与我们的模型 KGAT 相同，GAT 模型仅仅将基于核函数的注意力机制替换为基于点积的注意力机制。通过与 GAT 模型的表现进行对比能够进一步评估基于核函数的注意力机制的有效性。

4.4.3 实验设置

在此部分，我们具体介绍了相关的实验设置。事实验证任务参考了开放域问答的相关工作^[25]，采用了三步的流水线模型对待验证文本进行事实验证，其中包含文档检索，句子检索以及事实验证三个步骤。

文档检索。在文档检索这一步骤中，我们需要根据待验证文本检索出与之相关联的维基百科页面。与以前的工作保持相同^[27-29]，对于给定的待验证文本，我们首先利用 AllenNLP^[127] 中的成分句法分析 (Constituency Parser) 提取出所有可能表示实体的短语。然后，模型将这些解析出的短语用作查询，并通过在线的 MediaWiki

API^①进一步查找相关的维基百科页面。最后，我们将可信文本进行保留，形成检索到的文档集合^[29]。

句子检索。句子检索部分着重于从检索到的文档集合中选择出与待验证文本相关的句子。我们的实验设置了两种句子检索模型：第一种是基于自然语言推理模型 ESIM 的句子检索模型，第二种则为基于预训练语言模型 BERT 的句子检索模型。其中，基于 ESIM 的句子检索模型与以前的工作^[27,29]相同。基于 BERT 的句子检索模型是由我们设计训练完成的。我们参考了信息检索方面相关的工作^[32,72,112-113,128-129]，首先使用 BERT 来对待验证文本和检索到的文档集中的维基百科页面的证据句子进行拼接，并且使用 “[CLS]” 隐状态作为待验证文本和证据句子的表示。然后利用学习排序层（Learning-to-rank Layer）来将 “[CLS]” 隐状态映射成为排名分数。最后，我们采用了基于 Pairwise 的训练方式^[130]来训练我们的句子检索模型。值得一提的是，一些后续的相关工作^[131-132]也在他们的实验中沿用我们所提供的句子检索结果。

事实验证。在事实验证模型的训练过程中，我们采用 Adam 优化器^[75]用于模型参数的优化，批处理大小设置为 4，累积步长设置为 8。对于核函数数量的设定我们与前人工作^[32]一致，将其设置为 21。所有模型在训练过程均在开发集上使用事实验证标签的预测准确度（Label Accuracy）指标进行评估，并进行了两轮的训练。在训练集和验证集中，我们也将真正相关的证据加入其中，防止在训练的时候出现标签偏移现象。我们对每一个待验证文本保留了句子检索模块所提供的排名前五的证据，以此来对待验证文本进行事实验证。在事实验证过程中，我们的模型采用了不同的预训练语言模型进行效果评估，其中包含 BERT (Base)，BERT (Large) 和 RoBERTa (Large)^[50]。

4.4.4 事实验证效果

如表4.2所示，我们展示了在事实验证数据集 FEVER 上不同模型的表现。我们在此实验中提出了不同的测试场景，以此来验证我们模型的有效性。

首先，我们与实验中的主要基线模型 GEAR^[27]保持相同设置，即在句子检索阶段采用基于自然语言推理模型 ESIM 的句子检索模型，并且在事实验证阶段采用 BERT (Base) 作为句子的编码器。随后我们将句子检索模块替换为基于 BERT (Base) 的句子检索模型，并且比较了在事实验证阶段分别采用 BERT (Large) 和 RoBERTa (Large) 编码器时模型在事实验证任务上的效果。

在 FEVER 共享任务（1.0 版本）比赛中排名前三的模型以及基于预训练语言模型 BERT 的不同事实验证模型被用来验证我们的 KGAT 模型在事实验证任务上

① https://www.mediawiki.org/wiki/API:Main_page

表 4.2 事实验证准确性。

模型	验证集		测试集	
	Label Acc.	FEVER	Label Acc.	FEVER
Athene ^[29]	68.49	64.74	65.46	61.58
UCL MRG ^[105]	69.66	65.41	67.62	62.52
UNC NLP ^[104]	69.72	66.49	68.21	64.21
BERT Concat ^[27]	73.67	68.89	71.01	65.64
BERT Pair ^[27]	73.30	68.90	69.75	65.18
GEAR ^[27]	74.84	70.69	71.60	67.10
GAT (BERT Base) w. ESIM Retrieval	75.13	71.04	72.03	67.56
KGAT (BERT Base) w. ESIM Retrieval	75.51	71.61	72.48	68.16
SR-MRS ^[126]	75.12	70.18	72.56	67.26
BERT (Base) ^[28]	73.51	71.38	70.67	68.50
KGAT (BERT Base)	78.02	75.88	72.81	69.40
BERT (Large) ^[28]	74.59	72.42	71.86	69.66
KGAT (BERT Large)	77.91	75.86	73.61	70.24
KGAT (RoBERTa Large)	78.29	76.11	74.07	70.38

的有效性。与所有基线模型相比，我们的模型 KGAT 在所有的测试场景中都超过了全部的基线模型，达到了最好的效果。与以前的工作^[27,29]一样，我们采用了基于自然语言推理模型 ESIM 的句子检索模块，我们的模型 KGAT 在开发集和测试集上均优于基于图注意力神经网络的事实验证模型 GEAR^[27] 以及我们自己实现的基于图神经网络的事实验证模型 GAT。这个现象说明了相比较其他基于图的推理模型，我们设计的 KGAT 模型展示了其有效性。进一步，我们将句子检索模块替换为与基线模型^[28,126]一致的基于 BERT 的句子检索模型。在这个测试场景下，我们的模型 KGAT 的 FEVER 分数也比 BERT (Base)^[28] 高出近 1%，这显示出即使在不同句子检索模型下，我们的模型仍然能够保持一致的有效性。当我们将事实验证模块的句子编码器替换为 BERT (Large) 时，我们的模型 KGAT 也优于相应版本的基线模型^[28]。

与所有基线模型以及不同版本的 KGAT 模型相比，采用 RoBERTa (Large) 的 KGAT 模型表现最佳，进一步证明了我们的模型有着很好的泛化能力。CorefBERT^[132] 沿用了我们的 KGAT 模型框架，通过指导模型在上下文中显式建模并学习自然语言文本中存在的共指关系 (Coreference)，更好地提升如 BERT^[15] 和 RoBERTa^[50] 等预训练语言模型对于文本的推理能力，从而能够进一步提升事实验

证模型的准确性。

4.4.5 不同推理场景下模型效果评估

该实验研究了基于核函数的注意力机制在多证据和单证据推理场景下的有效性，此外我们也探究核函数在我们模型中的效果。

根据标注数据对于每个待验证文本所需要的事实验证证据的数量标注，可进行事实验证的实例（被标记为信息不足“NOT ENOUGH INFO”标签的实例除外）可以被分为两组。如果待验证文本需要多个证据才能推理出其事实验证标签，则认为此待验证文本需要多证据联合推理。根据待验证文本在事实验证过程中是否需要多条相关证据，我们进一步将事实验证场景划分为单证据推理场景和多证据推理场景，以此验证模型效果。单证据推理场景和多证据推理场景分别包含 11,372 (85.3%) 和 1,960 (14.7%) 个样本。

表 4.3 证据句子检索准确性评测结果。检索证据的效果是通过官方评价指标 **Precision**, **Recall** 和 **F1** 来进行评测^[103]。

	模型	Prec@5	Rec@5	F1@5	FEVER
验证集	ESIM	24.08	86.72	37.69	71.70
	BERT	27.29	94.37	42.34	75.88
测试集	ESIM	23.51	84.66	36.80	68.16
	BERT	25.21	87.47	39.14	69.40

在表4.3中，我们进一步比较了基于自然语言推理模型 ESIM 和预训练语言模型 BERT 的句子检索性能。如我们的实验结果所示，基于 BERT 的句子检索模型效果明显优于基于 ESIM 的句子检索模型。因此，更好的句子检索模型也有助于提高我们的模型 KGAT 的事实验证效果。不过，为了进行更公平的比较，我们与主要的基线模型 GEAR^[27] 保持一致，在基于 ESIM 的句子检索模型所提供的事实验证的基础上来进行下面的实验，进一步探究我们模型的有效性。

我们在实验中评估了我们模型 KGAT 的两个变体：第一个模型为 KGAT-Node，其保留了 Node Kernel 并将 Edge Kernel 替换为基于点积的注意力机制；第二个模型 KGAT-Edge 则与 KGAT-Node 相反，其保留了 Edge Kernel 并将 Node Kernel 替换为基于点积的注意力机制。上述几种模型在不同的推理场景下的事实验证效果如表4.4所示。我们通过比较没有核函数的图神经网络模型（GAT），仅具有 Node Kernel 机制的模型 KGAT-Node，仅具有 Edge Kernel 的模型 KGAT-Edge 以及完整的 KGAT 模型（KGAT-Full）来验证在不同推理场景下 KGAT 中不同的注意力机制的作用。

表 4.4 不同模型在多证据推理场景和单证据推理场景的事实验证效果。

推理场景	模型	Label Acc.	GFEVER	FEVER	
多证据推理	GEAR	66.38	n.a.	37.96	-0.25%
	GAT	66.12	84.39	38.21	-
	KGAT-Node	65.51	83.88	38.52	0.31%
	KGAT-Edge	65.87	84.90	39.08	0.87%
	KGAT-Full	65.92	85.15	39.23	1.02%
单证据推理	GEAR	78.14	n.a.	75.73	-1.69%
	GAT	79.79	81.96	77.42	-
	KGAT-Node	79.92	82.29	77.73	0.31%
	KGAT-Edge	79.90	82.41	77.58	0.16%
	KGAT-Full	80.33	82.62	78.07	0.65%

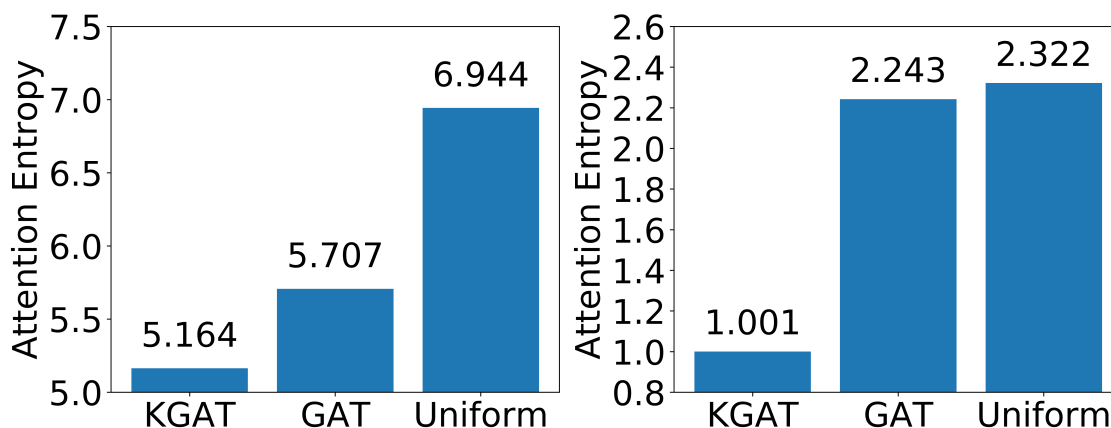
在基于单个证据和多证据的推理场景下，相比于 GAT，KGAT-Node 的性能均有 0.3% 以上的提升。不出所料，KGAT-Node 对 GFEVER 的效果并没有太大帮助。因为，在 GFEVER 指标的计算中，我们给出了足够验证输入文本的事实性证据，因此，在这种情况下，往往不需要对节点进行选择。这个现象说明了 KGAT-Node 着重于选择合适的证据并在节点的选择与读取时为其分配不同的权重。

不同于 KGAT-Node 的表现，在多证据推理和单证据推理两个场景下，KGAT-Edge 的性能分别比基线模型 GAT 高出 0.8% 和 0.1%，这个现象说明了 KGAT-Edge 的有效性主要来源于推理图中不同节点之间的推理信息传播，以此来更好地提升模型在多证据推理场景下的效果。

多证据和单证据推理场景从不同的方面评估了我们的 KGAT 模型的推理能力。单证据推理场景主要评价模型在选择最相关的证据并利用单证据进行推理的能力，其侧重于测试模型是否有能力在检索到的证据中根据真正相关的证据进行事实验证，进一步评估了模型的降噪能力。多证据推理是一个更困难、更复杂的场景，其需要模型来总结、归纳多条证据中必要的推理线索。多证据推理场景强调了模型在多证据联合推理中对证据的充分利用能力。KGAT-Node 在两种推理方案上均显示出一致的提升，这证明了证据选择的重要作用。另一方面，KGAT-Edge 在多证据联合推理场景中更为有效，因为 Edge Kernel 能够帮助模型更好地沿着推理图中节点的连边进行推理信息的传播。

4.4.6 核函数在事实验证中的作用

我们进一步通过本组实验证明核函数在 KGAT 模型中的作用。



(a) 证据传播的注意力 (Edge Attention)。

(b) 证据选择的注意力 (Node Attention)。

图 4.3 KGAT 注意力权重熵值。

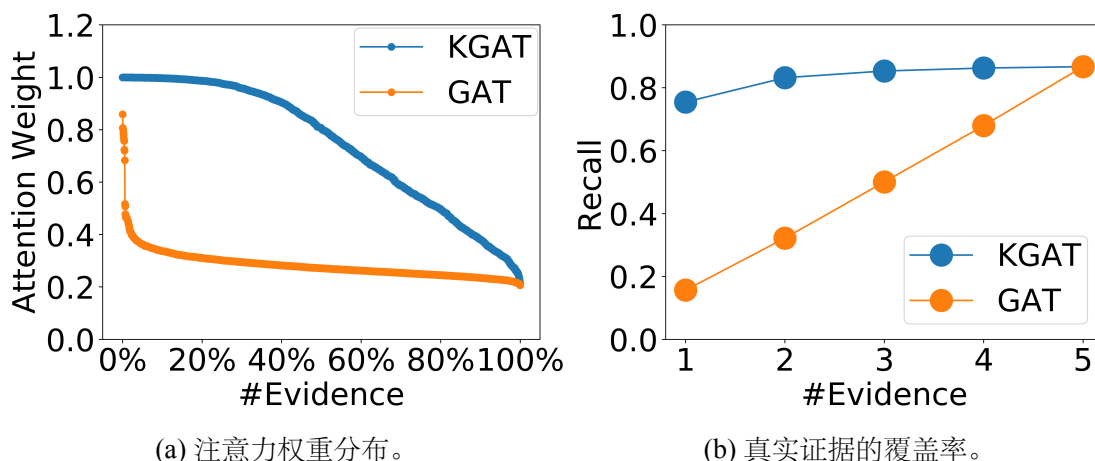
更集中的注意力机制。该实验通过研究注意力权重的熵值来探究基于核函数的注意力机制的有效性，注意力权重的熵值反映了不同模式下注意力机制学习到的权重是集中的还是分散的。

如图4.3所示，我们展示了不同的注意力机制的熵值，其中包含：KGAT 中基于核函数的注意力机制，GAT 中基于点积的注意力机制以及注意力权重均匀分布的注意力机制。我们比较了来自 KGAT 和 GAT 的证据传播时候的注意力机制 (Edge Attention) 和证据选择时候的注意力机制 (Node Attention) 的注意力权重的熵值。此外，均匀注意力权重分布的熵值也被引入到实验中以进行比较。在这里熵值越小，则代表注意力权重分布越集中。

如图4.3(a)所示，我们展示了在证据传播时候的注意力机制 (Edge Attention) 的熵值。与均匀分布的注意力机制相比，GAT 和 KGAT 两者的注意力机制都显示出更小的熵值，这说明了 GAT 和 KGAT 能够在基于点积和基于核函数的注意力机制下为某些重要的词语分配更多权重。与 GAT 中的基于点积的注意力机制相比，KGAT 的基于核函数的注意力机制能够将更多的权重分布到较少的词语上，并且能获得更小的熵值。

如图4.3(b)所示，我们进一步展示了在证据选择时候的注意力机制 (Node Attention) 的熵值。其中 GAT 的基于点积的注意力权重分布与均匀分布几乎相同，而 KGAT 则将对证据选择的注意力权重集中赋予一些证据上。正如下一个实验所示，基于核函数注意力机制能够将权重集中赋予一些正确的证据上，从而对检索到的证据进行消噪处理，这对于事实验证任务是十分有用的。

更准确的证据选择。该实验通过证据的注意力分布和真实证据的召回率来评估我们的模型 KGAT 在证据选择时候的注意力机制的有效性。其实验结果如图4.4所示。



(a) 注意力权重分布。

(b) 真实证据的覆盖率。

图 4.4 KGAT 和 GAT 的证据选择模块效果展示。

因为大多数待验证文本只需要单个证据即可对其进行事实验证，因此，我们首先从 KGAT 或 GAT 的证据图中获得对于每个证据节点的注意力权值，然后保留了对于每一个验证文本而言被分配最大注意力权值的证据节点，以此来研究证据选择注意力机制的有效性。如图4.4(a)所示，我们绘制了针对于每个待验证文本的证据句子集合中被赋予最高注意力权重的证据句子的注意力权值分布，我们的模型 KGAT 能够为其中一些证据分配更多的注意力权重，从而对这部分证据进行了重点关注。

随后，我们进一步通过对事实验证数据集中标注的真实证据的召回率来评估证据选择注意力机制的准确性。我们首先对每个待验证文本所有保留的证据依据证据选择注意力机制的权值进行排序。然后，我们保留不同排名深度的证据，并且计算模型对于真实证据的召回率，实验结果如图4.4(b)所示。我们的 KGAT 模型能够在排名靠前的位置上实现更高的召回率，仅保留排名第一的证据时，就能够覆盖近 80% 的真实证据，这个现象显示了我们的模型对于选择正确证据的能力。同时良好的证据选择效果也表明了基于核函数的文本匹配特征计算机制在句子检索阶段的潜力，本章工作重点讨论事实验证阶段的推理效果，我们将其保留以作为将来的探究工作。

更精细的证据传播。在本部分的最后一个实验中，我们分析研究了 KGAT 模型在证据传播时的注意力机制（Edge Attention）对于证据中每个词语的注意力权值分布情况，该分布用于测试模型对于每条证据中证据线索的归纳总结能力。

如图4.5所示，我们绘制了 KGAT 和 GAT 两个模型在证据传播时的注意力机制的注意力权重分布，一种来自于基于核函数的注意力机制，另一种则是来自于基于点积的注意力机制。我们给出了前 10% 的词语的注意力权重分布，其余词语的注意力权重分布遵循标准的长尾分布。我们的实验结果表明，基于核函数的注意

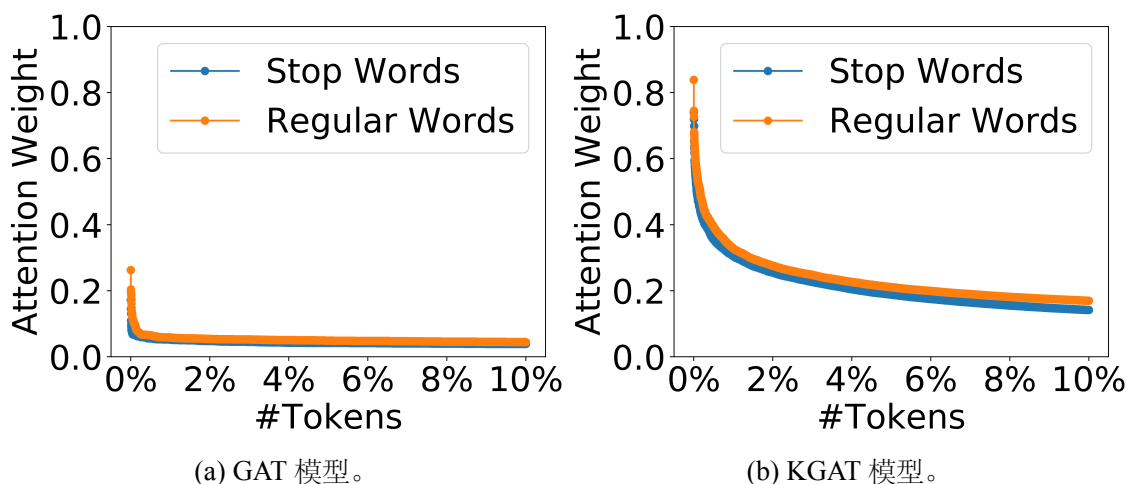


图 4.5 来自 GAT 和 KGAT 的证据传播时注意力机制对于证据中每个词语的注意力权重分布。

表 4.5 多证据联合推理样例分析。

待验证文本: *Al Jardine* is an *American rhythm guitarist*.

证据 (1) [Al Jardine] *Alan Charles Jardine* (born September 3, 1942) is an *American musician*, singer and songwriter who co-founded the Beach Boys.

证据 (2) [Al Jardine] *He is best known as the band's rhythm guitarist*, and for occasionally singing lead vocals on singles such as “Help Me, Rhonda” (1965), “Then I Kissed Her” (1965) and “Come Go with Me” (1978).

证据 (3) [Al Jardine] In 2010, Jardine released his debut solo studio album, *A Postcard from California*.

证据 (4) [Al Jardine] In 1988, Jardine was inducted into the Rock and Roll Hall of Fame as a member of the Beach Boys.

证据 (5) [Jardine] Ray Jardine American rock climber, lightweight backpacker, inventor, author and global adventurer.

预测标签: 支持 (SUPPORT)

真实标签: 支持 (SUPPORT)

力更加集中, 我们的 KGAT 模型相比较 GAT 模型能够将更多的注意力权重分配到较少的词语上面。然而, GAT 模型所采用的基于点积的注意力机制的注意力权重几乎平均分布在所有单词上, 基于点积的分散的注意力模式在先前的研究^[133]中同样被观察到。

如下一小节案例研究所示, 当整合来自多个片段的证据线索时, 我们的模型 KGAT 所采用的基于核函数的注意力机制能够提供更加细粒度且直观、有效的注意力模式。

KGAT AlJ ##ard ##ine is an American rhythm guitarist. [SEP] AlJ ##ard ##ine [SEP] He is best known as the band 's rhythm guitarist, and for occasionally singing lead vocals on singles such as " Help Me, R ##hon ##da " L ##RB 1965 R ##RB, " Then I Kiss ##ed Her " L ##RB 1965 R ##RB and " Come Go with Me " L ##RB 1978 R ##RB.

GAT AlJ ##ard ##ine is an American rhythm guitarist. [SEP] AlJ ##ard ##ine [SEP] He is best known as the band 's rhythm guitarist, and for occasionally singing lead vocals on singles such as " Help Me, R ##hon ##da " L ##RB 1965 R ##RB, " Then I Kiss ##ed Her " L ##RB 1965 R ##RB and " Come Go with Me " L ##RB 1978 R ##RB.

图 4.6 证据传播时的注意力机制 (Edge Attention) 对于证据中词语的注意力权重分配。

4.4.7 样例分析

如表格4.5所示, 一个在我们主要基线模型 GEAR^[27] 中所展示的样例被用来测试我们的 KGAT 模型效果。我们同样基于 ESIM 句子检索模型所检索到的证据语句作为事实验证证据, 其中前两个证据能够对待验证文本的事实正确性进行验证, 并且需要联合两条证据进行多证据推理。

如图4.6所示, 我们展示了第二条证据向第一条证据进行证据信息传播时候对于第二条证据中每个词语的注意力权重分布 ($\alpha_i^2 \rightarrow 1$), 其中基于点积的注意力机制 (GAT) 和基于核函数的注意力机制 (KGAT) 两种注意力机制的权重分布被画出。其中红色越深则表示注意力权值越高。

第一个证据证实“艾尔·贾丁是美国音乐家”, 但没有足够的信息证明“艾尔·贾丁是节奏吉他手”。KGAT 的基于核函数的证据传播注意力机制能够准确地从第二条证据中获取了第一条证据所需的附加信息, 节奏吉他手 (rhythm guitarist)。它能够有效地填补了缺失的信息并完善了推理链。有趣的是, 艾尔·贾丁 (Al Jardine) 也受到了更多关注, 这有助于验证第二条证据中的信息是否与待验证文本所描述的内容有关。这种基于核函数的注意力机制比 GAT 中的基于点积的注意力机制更加直观而且更加有效。基于点积的注意力机制几乎将注意力权重均匀地散布在所有词语上, 因此, 该种注意力机制很难解释多证据联合推理是如何进行的。这似乎是 Transformer^[133] 模型中基于点积的注意力机制的一个常见挑战。

4.5 本章小结

在本章中，我们基于图神经网络框架提出了基于核函数的多证据联合推理模型 KGAT，通过设计基于高斯核函数的两种注意力机制，证据传播注意力机制(Edge Attention)和证据选择注意力机制(Node Attention)从而实现细粒度的多证据联合推理以及更准确的证据选择。我们的实验表明，基于核函数的注意力机制能够实现更准确的事实验证分类效果以及提供更加直观且有效的注意力模式。我们的实验结果进一步表明，我们设计的两种注意力机制在事实验证中至关重要，并且在不同方面做出了贡献。标准的基于点积的注意力机制比较分散且难以解释，但通过利用基于核函数的注意力机制能够很好地减轻注意力分散的问题，并且显示了其更加直观、有效的注意力模式：基于核函数的证据选择注意力机制更多地关注与待验证文本有关的证据，证据传播注意力机制能够准确地根据一个证据节点从另一个证据节点收集必要的信息，以完成推理过程。将来，我们将在有效性方面和可解释性方面进一步研究基于核函数的注意力机制在神经网络中的应用。

第5章 面向专业领域的事实验证

在上一章中我们介绍了基于多事实证据的细粒度联合推理的事实验证模型。但是，在事实验证的过程中，其适用场景往往并不局限于通用领域，一些专业领域例如：政府公文校对、科学文章校对以及医学文章校对往往更加希望能够对将要发布的文章进行事实验证，从而保证撰写的文章具有客观性、科学性以及真实性。因此，为了能够在专业领域上充分发挥事实验证的效果，我们进一步通过对语言模型继续训练的方式进行研究，以此来强化语言模型对专业领域的术语理解，从而最终提升面向专业领域事实验证的效果。

在新型冠状病毒肺炎（COVID-19）流行期间，对于医学科学文本事实验证变得尤为重要。自动化的医学科学文本事实验证能够帮助用户对所看到的文章的事实真实性进行校验，从而在一定程度上避免类似于“5g 信号塔可以传播新冠肺炎病毒”，“可以考虑使用注射消毒剂来消灭新型冠状病毒肺炎”等，虚假的、违背科学事实的文本传播。但是，由于医学科学领域的事实验证训练数据十分匮乏，因此限制了事实验证模型的性能。在本章^①中，我们提出了两种面向专业领域的语言模型继续训练方法，用于提升模型在专业领域中对自然语言文本的理解能力，从而提升面向新型冠状病毒肺炎相关文本的事实验证的效果。

5.1 问题描述

在文本自动校对任务中，事实验证是其中一个重要部分，尤其是针对于一些专业领域。在新型冠状病毒肺炎流行期间，一些带有虚假信息的在线文章为了博人眼球，故意伪造虚假信息或故意编造违背科学的文章。此类文章经常被互联网用户关注并快速在网络上传播，从而引起了不必要的损失以及恐慌。因此，我们迫切地需要一个自动的事实校对系统来对文章中的内容进行事实层面的校对，以此来避免失实信息的传播所带来的不良后果。面向专业领域的事实校对工作已经引起了自然语言处理社区的广泛关注，相关工作也对多种信息载体进行了研究，例如：视频和文本，并对不同形式的信息进行事实层面的校对，从而进一步阻止错误信息的传播^[134-137]。

为了对新型冠状病毒肺炎有关的文本进行事实验证，Wadden 等人提出了面向

^① 本章主要工作以“Adapting Open Domain Fact Extraction and Verification to COVID-FACT through In-Domain Language Modeling”为题发表在 2020 年的国际学术会议 The Conference on Empirical Methods in Natural Language Processing (EMNLP 2020: Findings) 上。

医学科学领域的事实验证任务^[31]来对科学领域的文本进行事实验证，该任务将收集到的涵盖从基础科学到临床医学的各个领域的高质量论文作为可信语料库，从而对科学领域相关文本进行事实验证。**Wadden**等人证明了该任务能够很好地对与新型冠状病毒肺炎有关的文本进行事实验证。但是，与通用领域下的事实验证数据集 **FEVER**^[103]相比，**SCIFACT**的数据规模较小，在这样的小样本数据上训练事实验证模型限制了其在面向专业领域的事实验证任务上的效果。即使应用当前最广泛使用的预训练语言模型 **RoBERTa**^[50]，其事实验证的准确性 (**Precision**) 只能达到 46.6%^[31]，这样的事实验证准确性是难以让用户信任的。

正如我们前一章节所介绍的，事实验证模型的效果很大一部分取决于预训练语言模型对于自然语言的理解能力。因此，良好的事实验证准确率往往与预训练语言模型的自然语言的理解能力有着很大关系。预训练语言模型的成功往往来自于通过预训练方式从大规模语料库中学习词语的语义信息，进而强化了语言模型对于自然语言的理解能力^[138-139]。但是，针对于专业领域的语言模型，其通常会面临领域适应性问题，这是由于通用领域和专业领域的词语分布等不同所导致的。

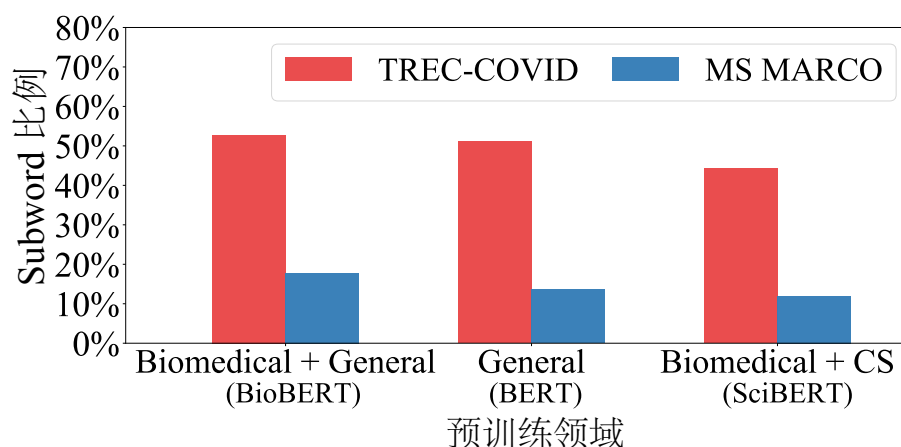


图 5.1 在新型冠状病毒肺炎有关的问题中单词被拆解成 Subwords 的单词比例。

现有的大多数预训练语言模型都将不常见的词划分为 **Subwords**^[140-141]，旨在缓解词表溢出 (**out-of-vocabulary**) 问题^[140]。如图5.1所示，对于面向新型冠状病毒肺炎相关的信息检索数据集 **TREC-COVID** 中的用户问题，其被划分为 **Subwords** 的比率显著高于通用领域信息检索数据集 **MS-MARCO**^[142] 中的问题。上述结果表明，现有的预训练语言模型大多都会将与新型冠状病毒 (COVID-19) 相关的术语视为未见过的词语，说明了现有的预训练语言模型可能对新型冠状病毒有关的术语的语义信息理解不充分，从而限制了预训练语言模型的效果。

除此之外，标注标签匮乏也限制了面向专业领域事实验证模型的效果。现有的针对新型冠状病毒肺炎的信息检索^[143]以及事实验证^[31]任务，由于需要雇佣一

些有专业背景知识的专家进行标注，因此其标注成本很高，导致了相关的标注数据的数量都很稀少。通过有限规模的数据训练模型，预训练语言模型不能够很好地通过下游任务来弥补预训练时候的领域差异问题，因此限制了预训练语言模型在专业领域事实验证任务上的效果。因此，我们基于现有的预训练语言模型进行继续训练，以此来学习到专业领域下的术语的语义信息，从而缓解领域适配问题。总结来说我们的工作有如下三个创新点：

(1) 为了解决预训练时候的领域差异问题，本章提出了两种用于事实证据提取和事实验证任务的面向专业领域的语言模型训练方法，基于掩码的语言模型训练方法 (Mask Language Model, MLM) 以及基于原理预测的语言模型训练方法 (Rationale Prediction, RP)，从而继续训练语言模型，让其学习到领域内术语的语义信息，进而得到面向专业领域的预训练语言模型。我们进一步将其与上一章所介绍的融合多事实证据的细粒度推理模型 KGAT^[26,132] 进行整合，并提出了面向科学领域的事实验证模型 (SciKGAT)，从而实现对新型冠状病毒肺炎相关的文本进行事实验证。

(2) 正如我们上一章所介绍的，事实验证任务往往采用了三步的流水线系统，文档检索，句子检索以及事实验证。因此，我们首先在文档检索这个子任务上对面向专业领域的语言模型效果进行评测。基于本工作设计的面向专业领域的预训练语言模型，我们在面向新型冠状病毒肺炎的信息检索比赛 (TREC COVID) 的第二轮无人工干预组中取得了第一名的成绩。从而证明了我们面向专业领域语言模型的有效性，帮助提升证据抽取部分的效果。

(3) 我们进一步评测了本工作设计的面向专业领域的预训练语言模型在面向专业领域事实验证上的效果。该工作所提出的面向科学领域的事实验证模型 SciKGAT 在专家撰写的科学事实验证数据集 SCIFACT^[31] 上的实验结果表明，SciKGAT 对于事实验证的准确性相比较基于 RoBERTa (Large) 的基线模型，其事实验证标签预测准确性 (Precision) 的绝对值提升了 30% (从 46.6% 到 76%)，F₁ 分数绝对值提升了 10%。显著的事实验证效果提升进一步证明了我们的面向专业领域的语言模型的有效性，同时也展示出我们在上一章所提出的融合多证据的细粒度推理模型 KGAT 在不同事实验证场景下一致的效果。我们的 SciKGAT 模型可以为低资源事实验证任务 (例如：新型冠状病毒肺炎领域) 提供一套解决方案。

5.2 相关工作

现有的事实抽取和验证任务 (Fact Extraction and Verification) 通常参考开放域问答的相关工作^[25] 采用三步流水线模型，实现对文本的事实验证，其包含了：文

档检索（摘要检索）、句子检索以及事实验证三个步骤^[31,103]。

事实验证任务往往需要联合多条证据进行推理，因此，很多工作将待验证文本与搜索到的多条证据进行拼接并用预训练语言模型对其进行编码表示，随后进行事实性标签预测^[31,104]。我们上一章所介绍的基于图神经网络的事实验证模型 KGAT，其首先将待验证文本与每条证据进行拼接作为证据节点，全连接所有证据节点形成推理图，进一步通过预训练语言模型得到每一个证据节点的表示。KGAT 设计了两种基于核函数^[112]的注意力机制进一步实现在一个推理图上进行细粒度的多证据联合推理，进而实现了高效的事实验证模型^[132]。

预训练语言模型^[15,50]通过预训练的方式指导语言模型从通用领域语料库中根据上下文学习词语的语义信息，进而提升模型对于自然语言文本的理解能力。因此，预训练语言模型有助于提高在通用领域下的事实验证^[26-28,110,132]以及信息检索的性能^[128,138,144]。

然而，在解决面向专业领域的自然语言处理问题时，预训练语言模型通常会面临着领域自适应问题。造成领域自适应问题的原因是由于不同领域中词汇分布的差异所导致的。虽然一些工作^[145-146]也在医学领域的语料库上训练语言模型，并且希望语言模型能够更好地理解医学相关术语的语义。但以新型冠状病毒肺炎领域为例^[147-148]，最早的相关出版物于 2019 年底出现。然而，像 COVID-19 这样的新兴医学术语，由于预训练语料库没有包含这样的新术语，因此即便是面向专业领域的语言模型^[145-146]，其对于新兴医学术语也是不熟悉的，从而限制了预训练语言模型在面向专业领域的事实验证任务中的效果。

5.3 模型框架

这一节描述了我们面向新型冠状病毒肺炎相关的事实验证模型 SciKGAT。其依旧采用三步的流水线模型^[24,26,103]进行事实证据抽取以及事实验证。随后介绍了我们对于预训练语言模型的继续训练方法。

5.3.1 事实验证流水线

给定待验证文本 c ，我们的目标是预测待验证文本的事实性标签 y （被分为支持、拒绝以及证据不足三类）。我们通常通过三个步骤的流水线模型来实现事实证据抽取以及事实验证：文档检索（摘要检索）、句子检索以及事实验证。

文档检索（摘要检索）。对于给定的待验证文本 c 和文档集合 $D = \{a_1, \dots, a_l\}$ ，我们的目标是通过以下步骤从其中检索出三个相关文档。在这里每一个文档为科学论文的摘要部分，因此，我们可以称其为摘要检索。

与基线模型^[31]相同，我们首先使用 TF-IDF 模型从文档集合 D 中检索出前 100 个与待验证文本相关的文档摘要。对于每一个摘要 a ，其包含了 k 条证据，因此，在这里我们将摘要 a 标记为如下形式 $a = \{e_1, \dots, e_k\}$ 。随后，我们将待验证文本 c 、摘要 a 所对应的标题 t 以及摘要 a 的全部文本进行拼接，进一步通过 BERT^[15] 进行编码，从而得到其隐状态表示 \mathcal{H}^e ：

$$\mathcal{H} = \text{BERT}([\text{CLS}] \circ c \circ [\text{SEP}] \circ t \circ a \circ [\text{SEP}]), \quad (5-1)$$

这里 \circ 代表了拼接操作。对于待验证文本 c 和摘要 a 的隐状态表示 \mathcal{H} ，其包含了来自待验证文本 c 和摘要 a 中所有词的隐状态表示。其中第 0 个词语 “[CLS]” 的隐状态表示被标记为 \mathcal{H}_0 。随后，待验证文本 c 和摘要 a 之间的相关性标签 y_a 的概率被计算出来：

$$p(y_a|c, a) = \text{softmax}_{y_a}(\text{MLP}(\mathcal{H}_0)), \quad (5-2)$$

在这里相关性标签 y_a 表示了待验证文本 c 和摘要 a 之间是相关的 ($y_a = 1$) 还是不相关的 ($y_a = 0$)。我们进一步根据待验证文本 c 和摘要 a 之间的相关性程度 ($p(y_a = 1|c, a)$) 对使用 TF-IDF 模型从文档集合 D 中检索的 100 个候选文档摘要进行重排序，其中排名最高的三个摘要被保留给后续的句子检索模块。

在这里我们可以采用一些先进的领域自适应算法^[72,149] 进行训练数据的增广以及选择，从而进一步提升文档检索的效果^①。我们进一步在面向新型冠状病毒肺炎的文档检索任务 TREC COVID^[143] 上进行模型评测，相关模型和代码开放到我们的信息检索工具包 OpenMatch^②中。

句子检索。 给定一个文档检索模型检索出的摘要 a ，句子检索模型希望能够从中找到能够对待验证 c 进行事实验证的证据句子。

类似于文档检索模型，对于给定的待验证文本 c 以及一个文档检索模型检索出的摘要 a 中的句子 e ，通过 BERT 模型我们可以得到二者 $(\langle c, e \rangle)$ 的词级别的隐状态表示：

$$\mathcal{H} = \text{BERT}([\text{CLS}] \circ c \circ [\text{SEP}] \circ e \circ [\text{SEP}]). \quad (5-3)$$

随后我们对待验证文本 c 与证据句子 e 的相关性标签 y_r 进行预测：

$$p(y_r|c, e) = \text{softmax}_{y_r}(\text{MLP}(\mathcal{H}_0)), \quad (5-4)$$

在这里相关性标签 y_r 表示了待验证文本 c 和证据句子 e 之间是相关的 ($y_r = 1$) 还是不相关的 ($y_r = 0$)。对于每一个摘要 a 来说，其中与待验证文本 c 相关的证据

① 相关工作主要在我们的预出版论文“CMT in TREC-COVID Round 2: Mitigating the Generalization Gaps from Web to Special Domain Search”中描述。

② <https://github.com/thunlp/OpenMatch>

句子 (即 $p(y_r = 0|c, e) < p(y_r = 1|c, e)$) 都被保留并形成了事实证据证据集合 E , 我们将其标记为 $E = \{e_1, \dots, e_q\}$ 。

事实验证。对于待验证文本 c 和检索到的相关证据句子集合 E , 事实验证模型期望能够根据检索到的证据来预测待验证文本的事实性标签 y 。我们在事实验证模块应用了我们上一章所介绍的 KGAT 模型来实现基于多证据联合推理的事实验证模块。通过预训练语言模型 BERT 我们可以得到待验证文本 c 与证据句子集合 E 中第 i 个证据 e_i 的词级别的隐状态表示 H^i 。随后事实验证标签 y 的概率可以被计算出来:

$$p(y|c, E) = \text{KGAT}(H^1, \dots, H^q). \tag{5-5}$$

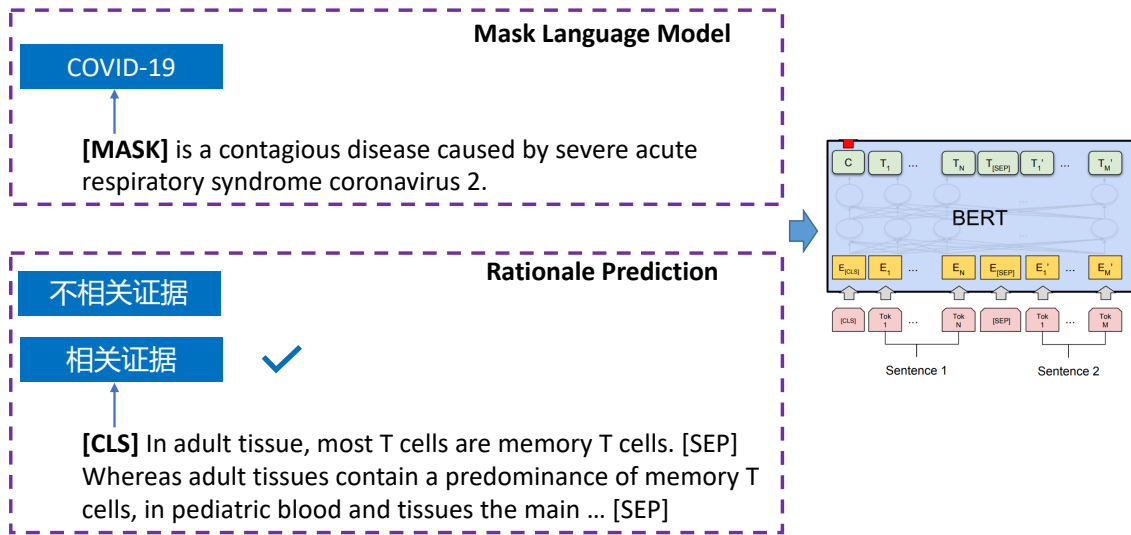


图 5.2 面向专业领域的语言模型继续训练方法示意图。

5.3.2 面向专业领域的语言模型继续训练

为了解决面向新型冠状病毒领域的事实验证中领域适配性问题, 如图5.2所示, 我们进一步提出了两种语言模型的继续训练方式, 通过继续训练方式让预训练语言模型更好地学习到专业领域下的相关术语的语义信息, 从而提升面向专业领域的事实验证效果。

对于面向新型冠状病毒肺炎的事实验证, 医学领域相关文本有助于理解医学词汇的语义信息^[145]。然而, 这些在医学领域数据集上训练的预训练语言模型往往会随着医学的发展或一种新病毒的出现而过时, 例如: COVID-19。利用专业领域下的训练数据对语言模型进行继续训练为解决这一问题提供了一种可能的方法。因此, 我们提出了两种面向专业领域的语言模型继续训练方式, 进一步用以提升

面向专业领域的事实验证的准确性。

基于原理预测的语言模型继续训练方法。我们首先提出了基于原理预测的语言模型继续训练方式 (**Rationale Prediction, PR**) 来训练语言模型,使其学习到新出现的专业领域中的术语的语义信息,从而提升预训练语言模型对于专业领域文本的理解能力。

对于给定的待验证文本 c 和标注数据集中给定的能够验证 c 的事实证据 e ,与之前所介绍的句子检索部分相类似,我们可以通过 BERT 对其进行编码表示,并用 “[CLS]” 作为句对 $\langle c, e \rangle$ 的表示,进一步我们可以预测出 $\langle c, e \rangle$ 的原理性标签 y_r 的概率 $p(y_r|c, e)$ 。随后我们利用标注数据集 SCIFACT^[31] 中的标注标签对其进行训练:

$$L_r(c, e) = \text{CrossEntropy}(p(y_r|c, e), y_r^*), \quad (5-6)$$

这里 y_r 与句子检索部分一致,表示了证据句子 e 与给定的待验证文本是相关 ($y_r = 1$) 还是不相关 ($y_r = 0$)。 y_r^* 则代表了标注数据集 SCIFACT^[31] 对 $\langle c, e \rangle$ 的相关性标注标签。随后,我们即可得到一个面向专业领域的预训练语言模型 BERT-RP。BERT-RP 用到了标注数据中的待验证文本与证据句子的相关性标签,因此,该预训练语言模型只能用于事实验证阶段。

基于掩码的语言模型继续训练方法。为了帮助模型更好地理解与新型冠状病毒肺炎相关的术语的语义信息,我们采用与 BERT^[15] 模型一样的基于掩码的语言模型继续训练方法 (**Mask Language Model, MLM**)。我们利用特殊标记符号 “[MASK]” 来随机替换专业领域语料中的词语,并要求模型选择适当的词语来填充被替换的词语。通过在专业领域的语料库上进行继续训练,语言模型可以从专业领域的语料库中阅读到有关术语的语义知识,并从中进一步理解新出现的专业领域术语,例如 COVID-19。通过在新型冠状病毒肺炎相关语料库上利用基于掩码的语言模型继续训练方法,能够使得语言模型更好地捕获此类新术语的上下文语义信息^[150]。从而提升语言模型在面向专业领域的事实验证的效果。

在基于掩码的语言模型继续训练方法中,我们使用了面向新型冠状病毒肺炎的医学研究论文数据集 (**COVID-19 Open Research Dataset**^①) 来继续训练语言模型。在此数据集的构建过程中,作者基于 AI2 搭建的文本过滤器对医学学术论文进行过滤。在该语料库中,截止到 2020 年之前大约有 86K 篇文章,它们都是与冠状病毒有关,而与新型冠状病毒肺炎无关;而在 2020 年之后有 54K 篇文章,其中大多数都与新型冠状病毒肺炎相关^[147]。

① <https://www.kaggle.com/allen-institute-for-ai/CORD-19-research-challenge>

5.4 实验

这一节中，我们将介绍在面向专业领域的事实验证任务中实验所用到的数据集，实验设置，基线模型以及最终的实验结果。

5.4.1 数据集

在本章的实验中，我们利用了最近发布的面向科学领域的事实验证数据集 SCIFACT^[31] 对模型效果进行测试，其数据集划分以及数据集统计信息如表5.1所示。SCIFACT 由 1,409 个被标注的待验证文本句子以及 5,183 篇科学论文组成。所有的待验证文本和 FEVER^[103] 类似，都被分类为支持，矛盾或信息不足。其训练集，开发集和测试集分别包含 809、300 和 300 组标注数据。相比较通用领域的事实验证数据集 FEVER 来说，SCIFACT 数据集标注数据数量较少。因此，官方提供的基线模型还使用通用领域下的事实验证数据集 FEVER^[103] 来训练事实验证流水线中的句子检索以及事实验证模块。FEVER 数据集则包括 185,455 个标注过的待验证文本句子以及 5,416,537 篇维基百科文档。

表 5.1 通用领域事实验证数据集 FEVER 以及面向科学领域数据集 SCIFACT 的训练集、验证集和测试集的划分。

数据集	数据集划分	# 支持	# 拒绝	# 信息不足
FEVER	训练集	80,035	29,775	35,639
	验证集	6,666	6,666	6,666
	测试集	6,666	6,666	6,666
SCIFACT	训练集	332	173	304
	验证集	124	64	112
	测试集	100	100	100

对于 SCIFACT，作者使用了 S2ORC^[151] 对数据集进行构建，这是一个由数百万篇科学文章构成的公开语料库。SCIFACT 随机抽取了从基础科学（如：Cell、Nature）到临床医学（如：JAMA、BMJ）等多个领域的知名期刊中的文章，并筛选了高质量的文章形成文档集合。最后，SCIFACT 通过论文中的引证关系，进一步通过人工标注构建数据集。作者发现，论文中的引证关系倾向于产生特定的、易于验证的文本。

此外，我们也在 SCIFACT 的文档检索子任务以及面向新型冠状病毒肺炎的文档级检索任务 TREC COVID^① 上进行实验。TREC COVID 第 2 轮的测试数据包括 2020 年 5 月 1 日版本的 CORD-19 文档集^[147]（其中包含了 59,851 篇与新型冠状病毒

① <https://ir.nist.gov/covidSubmit/index.html>

毒肺炎相关的论文)和生物医学专家编写的35个查询问题。在这些问题中,前30个问题已在第一轮中进行了标注。在实验中,我们使用TREC COVID第一轮的标注数据作为开发集(包含30个问题),问答数据集MS MARCO中医学相关的信息检索数据^[152]作为训练集(共78,895个标注问题),以此来训练文档检索模型。

5.4.2 基线模型

下面介绍我们在实验中所采用的基线模型。我们主要以SCIFACT^[31]中所提供的以预训练语言模型RoBERTa^[50]以及SciBERT^[145]为编码器的基线模型为主来评测不同的预训练语言模型在事实验证任务上的效果。

RoBERTa^[50]是面向通用领域的预训练语言模型,相比较BERT,其训练语料更多,采用了更优的训练策略,从而达到更好的效果。

SciBERT^[145]是一个使用生物医学(82%)以及计算机科学(12%)方向总共114万篇科技论文训练出来的预训练语言模型,该模型更加适用于面向科技领域的自然语言处理任务。

随后,我们利用所提出的语言模型训练方法分别对上述预训练语言模型继续训练,随后测试所得到的面向专业领域的语言模型在事实验证的两个子任务,证据搜索以及事实验证上的效果。此外,在事实验证子任务中,我们也采用上一章所介绍的KGAT模型作为事实验证中的多证据联合推理模型。

5.4.3 实验设置

在所有实验中,我们选用SciBERT, RoBERTa (Base)和RoBERTa (Large)三种预训练语言模型^[50,145]进行实验,并实现与各个预训练语言模型所对应的,基于原理预测的继续训练(PR)以及基于掩码的继续训练(MLM)的面向专业领域的语言模型。预训练语言模型的实现继承了Huggingface Transformers^[74]工具包给出的实现。在实验中我们采用Adam优化器^[75]进行神经网络参数优化。下面我们分别介绍在事实验证流水线中三个步骤:文档检索、句子检索以及事实验证上的实验设置。

文档检索。对于文档检索模块,我们跟以前的工作^[152]保持一致,使用MS MARCO^[142]语料库中与医学相关的问题文档的相关性标注对语言模型进行训练,使文档检索模块适用于面向医学领域的文档检索任务。我们在训练过程中将语言模型的最大输入长度设置为256,学习率设置为 $2e-5$,批处理大小设置为8,累计步长设置为4。

句子检索。对于句子检索模型,我们保持与基线模型^[31]相同的设置。

事实验证。对于事实验证,前一章所介绍的多证据联合推理模型KGAT也被

用来评测我们提出的语言模型的效果，我们在训练过程中将最大长度设置为 256，学习率设置为 $2e-5$ ，批处理大小设置为 8，累计步长设置为 4。KGAT 中其他参数的设置与之前相同。

表 5.2 SciKGAT 在面向专业领域的事实验证任务上的效果。

模型	验证集						测试集					
	句子级别			文档级别			句子级别			文档级别		
	P	R	F1	P	R	F1	P	R	F1	P	R	F1
Baselines												
SciBERT	45.78	38.52	41.84	51.93	44.98	48.21	-	-	-	-	-	-
RoBERTa	46.51	38.25	41.98	53.30	46.41	49.62	38.6	40.5	39.5	46.6	46.4	46.5
SciKGAT												
KGAT	57.07	31.97	40.98	72.73	38.28	50.16	-	-	-	-	-	-
SciKGAT (w. A)	42.07	47.81	44.76	47.66	58.37	52.47	40.50	48.38	44.09	47.06	57.66	51.82
SciKGAT (w. AR)	50.00	47.81	48.88	53.15	56.46	54.76	41.67	45.95	43.70	47.47	54.96	50.94
SciKGAT (Full)	74.36	39.62	51.69	84.26	43.54	57.41	61.15	42.97	50.48	76.09	47.30	58.33

5.4.4 面向专业领域的事实验证流水线模型效果

SciKGAT 的总体性能如表5.2所示，我们所采用的主要基线模型 RoBERTa 使用 TF-IDF 进行文档检索，使用 RoBERTa (Large) 进行句子检索以及基于多证据联合推理的事实验证。我们逐步加入 SciKGAT 的不同模块来评估模型的有效性。在这里 RoBERTa 代表了 RoBERTa (Large)。SciKGAT (w. A) 代表了只采用 SciKGAT 的文档检索部分。SciKGAT (w. AR) 代表了只采用 SciKGAT 的文档检索部分以及句子检索部分。在本部分实验中我们采用了基于掩码的语言模型继续训练方式得到面向专业领域的预训练语言模型。

相比较我们的基线模型，在 SciKGAT (w.A) 和 SciKGAT (w.AR) 两个模型中，我们采用与基线模型一致的基于 RoBERTa (Large) 模型实现事实验证模块。实验结果表明，在文档级以及句子级的信息检索场景下，面向专业领域的预训练语言模型可以从文档检索和句子检索两个层面分别增强证据检索的准确性，进而提升事实验证的效果。

应用全部的 SciKGAT 的模块后，相比较基线模型，SciKGAT (Full) 在事实验证任务上的准确性 (Precision) 的绝对值提升了 30%， F_1 分数绝对值提升了 10%。事实验证的准确性方面的显著提升证明了我们的模型 SciKGAT 能够为医学科学有关的文本提供可信的事实验证结果。在下面的实验中，我们具体分析面向专业领域的预训练语言模型在事实验证流水线中每个模块的作用。

表 5.3 在 SCIFACT 验证集上文档检索效果。

文档检索	排序准确性		文档级别			句子级别		
	Hit one	Hit all	P	R	F1	P	R	F1
TF-IDF	84.67	83.33	53.30	46.41	49.62	46.51	38.25	41.98
w. SciBERT	94.67	93.00	48.18	56.94	52.19	42.09	47.27	44.53
w. SciBERT (MLM)	95.33	93.67	47.66	58.37	52.47	42.07	47.81	44.76

表 5.4 TREC COVID^[143] 文档检索比赛中第二轮无人工干预组各机构排名。

研究机构	NDCG@10	P@5	RBF(p=.5)	bpref	MAP
清华大学 (中国)	0.6772	0.7600	0.7520	0.5096	0.3115
马克斯普朗克研究所 (德国)	0.6677	0.7771	0.7305	0.4609	0.2946
爱荷华大学 (美国)	0.6382	0.7657	0.7037	0.4869	0.2845
Salesforce (美国)	0.6320	0.7086	0.7153	0.4414	0.3000
滑铁卢大学 (加拿大)	0.6250	0.7314	0.6634	0.4876	0.2880
Sabir Research (美国)	0.6161	0.7086	0.6746	0.4704	0.2269
CogIR (美国)	0.6131	0.7086	0.6456	0.4222	0.2590
Risklick (瑞士)	0.5998	0.7143	0.6773	0.4203	0.2312

5.4.5 语言模型在文档检索任务上的效果

我们首先比较了面向专业领域的预训练语言模型在对检索到的文档进行重排序的效果，如表5.3所示。在这里我们采用了基于掩码的继续训练方式实现了面向专业领域的预训练语言模型。我们设计了两步的文档检索方式：采用 TF-IDF（或 BM25）模型进行初筛，后采用神经网络模型进行重排序。

在训练文档检索重排序模块的时候，我们跟以前的工作^[152]一致，使用 MS MARCO^[142] 语料库中与医学相关的问题文档相关性标注进行模型训练。我们使用大规模的监督语料并能够显著提升事实验证任务中文档检索的效果，一致的实验现象也在相关文章^[149,152]中被观察到。进一步我们使用了面向专业领域的预训练语言模型在 MS MARCO 语料库上进行微调，我们发现面向专业领域的预训练语言模型能够提升相关文档的命中率。随着文档检索效果的提升，相应的事实验证的召回率（Recall）会有显著的提升，进一步提升模型的效果（F1 值）。

同时我们也参加了美国国家标准与技术研究院（National Institute of Standards and Technology, NIST）所举办的针对于新型冠状病毒肺炎的文档级检索比赛 TREC COVID。如表5.4所示，我们在第二轮的无人工干预组中达到了第一名的水平^①，

① <https://castorini.github.io/TREC-COVID/round2/>

表 5.5 在面向新型冠状病毒信息检索任务 TREC COVID 中检索模型的消融实验。

模型	第一轮（验证集）		第二轮（测试集）	
	NDCG@10	P@5	NDCG@10	P@5
SciBERT	0.6598	0.7733	0.5828	0.6629
SciBERT (MLM)	0.6775	0.7400	0.5880	0.6800
SciBERT-Full (MLM)	0.7470	0.9000	0.6953	0.7829

展示了我们的模型在面向专业领域文档检索任务上良好的效果。我们依旧采用了同样的实验设置，在重排序阶段我们依旧采用了面向专业领域的预训练语言模型 SciBERT (MLM)，并在其基础上使用 MS MARCO^[142] 中与医学相关的问题文档标注进行模型训练，实验结果如表5.5所示。一致的实验结果证明了我们的语言模型继续训练方法能够提升面向专业领域信息检索的效果。此外，我们在文档检索的第一步应用了 Dense Retrieval^[153-155] 增强对于检索文档的召回率，并采用一些先进的数据筛选方式^[72] 进行数据筛选并实现了 SciBERT-Full 模型，进一步提升了文档检索的效果，为进一步提升事实验证效果提供了潜在的研究方向。

5.4.6 语言模型在句子检索任务上的效果

表 5.6 在 SCIFACT 验证集上句子检索效果。

句子检索	检索准确性			句子级别			文档级别		
	P	R	F1	P	R	F1	P	R	F1
SciBERT	36.90	65.03	47.08	43.22	46.99	45.03	48.94	55.02	51.80
SciBERT (MLM)	43.73	60.93	50.91	50.00	47.81	48.88	53.15	56.46	54.76
RoBERTa-Base	37.56	61.48	46.63	43.64	45.90	44.74	46.06	53.11	49.33
RoBERTa-Base (MLM)	29.82	61.75	40.21	41.45	48.36	44.64	45.02	54.07	49.13
RoBERTa-Large	36.78	64.21	46.77	42.07	47.81	44.76	47.66	58.37	52.47
RoBERTa-Large (MLM)	38.44	63.11	47.78	42.93	46.45	44.62	47.03	53.11	49.89

在本实验中，我们进一步探究了面向专业领域的语言模型在句子检索任务中的作用，实验结果如表5.6所示。

我们采用基于掩码的语言模型的训练方式分别在 SciBERT, RoBERTa (Base) 和 RoBERTa (Large) 三种预训练语言模型上进行实验来验证其效果。在实验中，面向科学领域的语言模型 SciBERT 通过继续训练后有了显著的提升，这个现象证明了通过继续训练的方式能够帮助面向专业领域的预训练语言模型更新所学习到的知识。然而，对于通用领域下的预训练语言模型 RoBERTa，我们的继续训练方

式似乎并无效果，可能的原因是我们的继续训练语料规模较小且都是面向冠状病毒相关的文档，因此，很难通过我们的继续训练方式使得通用领域的预训练语言模型在句子检索这个任务中适配到专业领域。

表 5.7 不同模型在 SCIFACT 开发集上的事实验证效果。

事实验证模型	文档级别			句子级别		
	P	R	F1	P	R	F1
SciBERT (Concate)	36.92	45.93	40.94	36.55	38.25	37.38
w. KGAT (SciBERT Init)	58.99	39.23	47.13	51.61	34.97	41.69
w. KGAT (SciBERT-RP Init)	66.38	36.84	47.38	60.10	33.33	42.88
w. KGAT (SciBERT-MLM Init)	65.32	38.76	48.65	56.00	34.43	42.64
RoBERTa-Base (Concate)	44.50	46.41	45.43	42.72	36.89	39.59
w. KGAT (RoBERTa Init)	68.87	34.93	46.35	61.05	31.69	41.73
w. KGAT (RoBERTa-RP Init)	67.48	39.71	50.00	61.19	36.61	45.81
w. KGAT (RoBERTa-MLM Init)	67.19	41.15	51.04	60.35	37.43	46.21
RoBERTa-Large (Concate)	53.15	56.46	54.76	50.00	47.81	48.88
w. KGAT (RoBERTa Init)	72.39	46.41	56.56	62.87	40.71	49.42
w. KGAT (RoBERTa-RP Init)	83.33	43.06	56.78	73.47	39.34	51.25
w. KGAT (RoBERTa-MLM Init)	84.26	43.54	57.41	74.36	39.62	51.69

5.4.7 语言模型在事实验证任务上的效果

在这个实验中，我们评估了面向专业领域的语言模型在事实验证任务上的效果，其中上一章所介绍的基于多证据细粒度联合推理模型 KGAT 也被加入到实验中进行比较。

如表5.7所示，在实验中，我们比较了 SciBERT 和 RoBERTa 两种预训练语言模型以及相对应的面向专业领域的语言模型，即 MLM 模型（mask language model training）和 RP 模型（rationale prediction training），在事实验证任务上的作用。我们在实验中使用基于多证据细粒度推理的事实验证模型 KGAT^[26,132] 对面向专业领域的预训练语言模型 MLM（基于掩码继续训练的语言模型）和 RP（基于原理预测继续训练的语言模型）进行了评估。

我们可以发现，相比较将证据全部拼接的基线模型，KGAT 模型能够借助于其所设计的细粒度推理机制显著地提升模型的标签预测的准确性，从而提升模型在事实验证任务上的性能，为用户提供更加精准的事实验证标签预测结果。在加入我们的面向专业领域的预训练语言模型后，相应模型的效果进一步被提升，从

而证明了在面向专业领域的事实验证任务中强化语言模型对于专业领域术语的语义理解的重要性。相比较 RP 模型，基于掩码的语言模型继续训练方式受益于其语料规模能够更显著地提升语言模型在专业领域文本推理上的效果，从而进一步提升模型在事实验证任务上的性能。

表 5.8 事实验证结果样例分析。

待验证文本: Basophils counteract disease development in patients with systemic lupus erythematosus (SLE). [嗜碱性粒细胞可抵抗系统性红斑狼疮 (SLE) 患者的疾病发展。]

证据 1: ... *basophils* and IgE autoantibodies amplify autoantibody production that *leads to lupus nephritis* ... [嗜碱性粒细胞和 IgE 自身抗体会强化自身抗体的产生，从而导致狼疮性肾炎]

证据 2: *Individuals with SLE also have elevated* serum IgE, self-reactive IgEs and *activated basophils* that ... [系统性红斑狼疮 (SLE) 患者的血清 IgE、自身反应性 IgE 和激活的嗜碱性粒细胞也升高]

SciKGAT: 拒绝 (REFUTES) RoBERTa: 证据不足 (NOT ENOUGH INFO)

待验证文本: In adult tissue, most T cells are memory T cells. [在成人体内，大多数 T 细胞是记忆性 T 细胞。]

证据 1: *Whereas adult tissues contain a predominance of memory T cells*, in pediatric blood and tissues the main subset consists of naive recent thymic emigrants ... [成人体内记忆性 T 细胞占多数，相反在儿童的血液和身体组织中，主要的细胞亚群是从胸腺迁移过来的幼稚细胞。]

SciKGAT: 支持 (SUPPORTS) KGAT: 拒绝 (REFUTES)

5.4.8 样例分析

如表5.8所示，我们在面向科学领域的事实验证数据集 SCIFACT 的开发集上挑选出的两个样例，以此来进一步证明我们提出的面向专业领域的事实验证模型 SciKGAT 的有效性。实验中所有模型均使用 RoBERTa (Large) 来实现。被标注为红色的部分是每条证据中可以对待验证文本进行事实验证的证据。

在第一个例子中，对于待验证文本“嗜碱性粒细胞可抵抗系统性红斑狼疮 (SLE) 患者的疾病发展。”，证据 1 和证据 2 表明了嗜碱性粒细胞可导致系统性红斑狼疮以及系统性红斑狼疮患者的嗜碱性粒细胞会增加，这些线索都与待验证文本相矛盾。基于拼接的事实验证模型 RoBERTa 未能利用给定的证据验证这一文本是违背事实的，而 SciKGAT 做出了正确的预测，从而证明了基于多证据细粒度推理模型 KGAT 的有效性。在第二个例子中，检索到的证据 1 表明成人的记忆 T 细胞最多。SciKGAT 通过对这些医学短语的识别和理解，正确地预测了待验证文本的事实验证标签，并显示了其有效性。

5.5 本章小结

在本章中，我们提出了面向专业领域的语言模型继续训练方法，该方法能进一步将专业领域知识迁移到面向专业领域的事实验证任务中来。我们的实验表明，我们的事实验证流水线模型能显著提高事实验证模型的性能。我们的分析表明，我们的模型通过继续训练的方式使其学习到专业领域的相关知识，进而增强了语言模型的推理能力。

第6章 总结与展望

随着互联网的不断发展，在线文本数量不断增加，如何能够对文本进行自动化校对是一个十分重要的任务。自动文本校对方法旨在帮助人们实现文本自动校对系统，其中包含了两个部分，语法错误校对以及事实验证，从而保证文本的正确性以及真实性。为了对文本的语法错误以及事实错误进行识别，我们可以通过整合一些富信息对文本进行校对，例如语言学知识、世界知识以及领域知识，并依靠预训练语言模型的语言建模能力以及推理能力进一步实现一个高效的文本自动校对工具。本文针对现有的文本校对中**信息层、算法层以及应用层**所面临的**缺乏语法错误校对证据、缺少融合多语法改错结果的语法错误检查模型、缺少融合多事实证据的事实错误检查模型以及难以对专业领域文本进行事实方面的校验**四个问题，系统性地开展了四个工作，在语法错误校对以及事实错误校对任务上取得了良好的效果。

6.1 主要贡献

本文的主要贡献包括以下四点：

- 针对于英文语法校对中语法改错预训练方式的选择以及调参，本文讨论了**语法改错算法的框架和语法改错证据获取算法**。本工作探究了现有的通用的预训练语言模型以及针对于语法改错任务的预训练模型的效果，并通过调试不同参数得到当前最好的基于生成的语法改错模型。此外，为了能够充分地利用标注数据，我们参考了前人的工作，提出了一种基于强化学习的数据筛选方法，通过对语言学习者网站上的数据进行筛选，进一步提升了语法改错的效果。最终利用训练好的语法改错模型通过柱搜索解码方式得到若干个语法改错结果，从而辅助语法错误校对。
- 人们在实际进行语法校对的时候总是希望得到有关的语法改错的提示，以此来辅助语法错误修改。受到了人类行为的启发，为了充分地利用语法改错模型所提供的语法改错线索，本文提出了**融合多语法改错证据的英文语法校对算法**。通过本算法能够充分地利用上一工作所提供的语法改错线索，进而提升语法校对能力。本文在四个通用语法改错数据集上进行了实验，实验结果表明该算法在语法错误检查以及语法改错质量评估两个任务上均优于其他基线模型，同时利用该算法能够通过语法改错模型柱搜索结果进行重排序进一步提升语法改错效果。

- 针对于事实验证任务，本文提出了**融合多事实证据的细粒度联合推理的事实验证算法**。由于在事实验证过程中，相关的事实证据检索都是通过信息检索模型完成的，因此不可避免地会引入额外的噪音。此外，在检索到的句子中也仅有一部分对验证当前文本的语义真实性以及完整性起到作用。因此本工作希望能够利用多个事实验证证据进行细粒度联合推理，进一步提升模型在事实验证层面的推理能力。该方法在事实验证通用数据集上超过了其他的基线模型，证明了其良好的推理能力以及事实验证效果。
- 在事实验证任务中，其处理的文本很可能是面向专业领域的文章，本文提出了**面向专业领域的事实验证算法**，进一步提升事实验证模型在面向专业领域的事实验证任务上的效果。该算法提出了两种面向专业领域的预训练语言模型的继续训练方法，以此提升预训练语言模型模型在专业领域上对文本的理解能力。该方法在面向医学科学领域的事实验证数据集上效果优于通用领域预训练语言模型，进而证明了其有效性。

6.2 未来工作展望

本文针对融合富信息的英文文本自动校对展开研究，并尝试通过引入语言知识、世界知识以及领域知识等富信息来提升文本自动校对的能力，这里列举一些可能的未来研究方向，希望能对相关研究者有所启发：

- **融合更加多样的语法错误校对证据**。针对于文本自动校对任务，我们可以提升在语法错误校对阶段的语法错误校对证据的多样性，例如：可以考虑已有的相关例句，亦或是通过采用诸如 **Temperature Sampling** 以及 **Top-K Sampling** 等解码方式增强解码过程的多样性，进而产生更加多样的语法改错结果。通过不同来源或不同的解码方式得到的例句或者语法改错结果来进一步提升语法错误校对证据的多样性，从而提升语法错误校对模型的效果。
- **对事实错误进行修正**。现有的事实验证模型还停留在对于文本进行事实错误检查阶段，即根据搜索到的事实验证证据来判断待验证文本是否符合事实知识。受益于现有的预训练语言模型，如 T5^[63] 和 BART^[18]，的强大的生成能力以及 Dense Retrieval^[153,155] 高效的信息检索能力，实现一个端到端的检索、事实错误修正模型是一个十分重要的研究方向。
- **通过预训练语言模型提升在不同场景下的文本校对能力**。现有的预训练语言模型虽然对语言知识有着很好的建模效果并且展示出其对于自然语言文本的理解能力，然而对于语法错误校对以及事实错误校对来说，其对于语言模型的能力有着不同的需求。针对语法层面，语言模型往往需要注意词语的词

形、词性、搭配词等等，而针对于语义（事实）层面，其往往关注人名、时间以及地点等信息。因此，如何能够针对两个任务的不同需求分别提升语言模型的效果是一个重要的研究问题。

- **面向篇章级文本自动校对。**现有的工作往往是在句子级别进行文本的自动校对，然而很多文本是以篇章形式存在。因此，对于每一个句子的上下文的连词使用是否正确，转折是否合适，上下文是否出现了语义矛盾，是否存在跑题等问题的校对工作也是十分重要的。然而此部分工作开展的较少，因此也需要进一步研究。

参考文献

- [1] 何晓霞. 图书编校质量问题及其对策 [J]. 硅谷, 2015, 001(020): 66-67.
- [2] Wang Y, Wang Y, Liu J, et al. A comprehensive survey of grammar error correction[J]. arXiv preprint arXiv:2005.06600, 2020.
- [3] 张仰森. 统计语言建模与中文文本自动校对技术 [M]. 2017.
- [4] 张仰森, 唐安杰, 张泽伟. 面向政治新闻领域的中文文本校对方法研究 [J]. 中文信息学报, 2014, 28(6): 79-84.
- [5] Daudaravicius V, Banchs R E, Volodina E, et al. A report on the automatic evaluation of scientific writing shared task[C]// Proceedings of the 11th Workshop on Innovative Use of NLP for Building Educational Applications. 2016: 53-62.
- [6] Hitomi Y, Tamori H, Okazaki N, et al. Proofread sentence generation as multi-task learning with editing operation prediction[C]// Proceedings of IJCNLP. 2017: 436-441.
- [7] 刘建明. 宣传舆论学大辞典 [M]. 1993.
- [8] Ng H T, Wu S M, Briscoe T, et al. The CoNLL-2014 shared task on grammatical error correction[C]// Proceedings of the Eighteenth Conference on Computational Natural Language Learning: Shared Task. 2014: 1-14.
- [9] Rei M. Semi-supervised multitask learning for sequence labeling[C]// Proceedings of ACL. 2017: 2121-2130.
- [10] Rei M, Søgaard A. Jointly learning to label sentences and tokens[C]// Proceedings of AAAI. 2019: 6916-6923.
- [11] Dale R, Kilgarrieff A. Helping our own: The HOO 2011 pilot shared task[C]// Proceedings of the 13th European Workshop on Natural Language Generation. 2011: 242-249.
- [12] Dale R, Anisimoff I, Narroway G. HOO 2012: A report on the preposition and determiner error correction shared task[C]// Proceedings of the Seventh Workshop on Building Educational Applications Using NLP. 2012: 54-62.
- [13] Ng H, Wu S, Wu Y. Ch. hadiwinoto, and j. tetreault. 2013. the conll-2013 shared task on grammatical error correction[J]. Proceedings of CoNLL.
- [14] Hochreiter S, Schmidhuber J. Long short-term memory[J]. Neural computation, 1997, 9(8): 1735-1780.
- [15] Devlin J, Chang M W, Lee K, et al. BERT: Pre-training of deep bidirectional transformers for language understanding[C]// Proceedings of NAACL. 2019: 4171-4186.
- [16] Kaneko M, Komachi M. Multi-head multi-layer attention to deep language representations for grammatical error detection[J]. Computación y Sistemas, 2019, 23(3).
- [17] Vaswani A, Shazeer N, Parmar N, et al. Attention is all you need[C]// Proceedings of NIPS. 2017: 5998-6008.

-
- [18] Lewis M, Liu Y, Goyal N, et al. BART: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension[C]// Proceedings of ACL. 2020: 7871-7880.
- [19] Kiyono S, Suzuki J, Mita M, et al. An empirical study of incorporating pseudo data into grammatical error correction[C]// Proceedings of EMNLP. 2019: 1236-1242.
- [20] Chen M, Ge T, Zhang X, et al. Improving the efficiency of grammatical error correction with erroneous span detection and correction[C]// Proceedings of EMNLP. 2020: 7162-7169.
- [21] Yannakoudakis H, Rei M, Andersen Ø E, et al. Neural sequence-labelling models for grammatical error correction[C]// Proceedings of EMNLP. 2017: 2795-2806.
- [22] Zhao W, Wang L, Shen K, et al. Improving grammatical error correction via pre-training a copy-augmented architecture with unlabeled data[C]// Proceedings of NAACL. 2019: 156-165.
- [23] 桂红星, 陈晖. 报纸重大差错的成因及防堵 [J]. 新闻前哨, 2006, 000(008): 35-36.
- [24] Thorne J, Vlachos A, Cocarascu O, et al. The fact extraction and VERification (FEVER) shared task[C]// Proceedings of the First Workshop on Fact Extraction and VERification (FEVER). 2018: 1-9.
- [25] Chen D, Fisch A, Weston J, et al. Reading Wikipedia to answer open-domain questions[C]// Proceedings of ACL. 2017: 1870-1879.
- [26] Liu Z, Xiong C, Sun M, et al. Fine-grained fact verification with kernel graph attention network[C]// Proceedings of ACL. 2020: 7342-7351.
- [27] Zhou J, Han X, Yang C, et al. GEAR: Graph-based evidence aggregating and reasoning for fact verification[C]// Proceedings of ACL. 2019: 892-901.
- [28] Soleimani A, Monz C, Worring M. BERT for evidence retrieval and claim verification[J]. arXiv preprint arXiv:1910.02655, 2019.
- [29] Hanselowski A, Zhang H, Li Z, et al. UKP-athene: Multi-sentence textual entailment for claim verification[C]// Proceedings of the First Workshop on Fact Extraction and VERification (FEVER). 2018: 103-108.
- [30] Zhong W, Xu J, Tang D, et al. Reasoning over semantic-level graph for fact checking[C]// Proceedings of ACL. 2020: 6170-6180.
- [31] Wadden D, Lin S, Lo K, et al. Fact or fiction: Verifying scientific claims[C]// Proceedings of EMNLP. 2020: 7534-7550.
- [32] Qiao Y, Xiong C, Liu Z, et al. Understanding the behaviors of bert in ranking[J]. arXiv preprint arXiv:1904.07531, 2019.
- [33] Naber D, et al. A rule-based style and grammar checker[J]. 2003.
- [34] Gamon M, Leacock C, Brockett C, et al. Using statistical techniques and web search to correct esl errors[J]. Calico Journal, 2009, 26(3): 491-511.
- [35] Mizumoto T, Komachi M, Nagata M, et al. Mining revision log of language learning SNS for automated Japanese error correction of second language learners[C]// Proceedings of IJCNLP. 2011: 147-155.

-
- [36] Dahlmeier D, Ng H T, Wu S M. Building a large annotated corpus of learner English: The NUS corpus of learner English[C]// Proceedings of the Eighth Workshop on Innovative Use of NLP for Building Educational Applications. 2013: 22-31.
- [37] Bryant C, Felice M, Andersen Ø E, et al. The BEA-2019 shared task on grammatical error correction[C]// Proceedings of the Fourteenth Workshop on Innovative Use of NLP for Building Educational Applications. 2019: 52-75.
- [38] Brockett C, Dolan W B, Gamon M. Correcting ESL errors using phrasal SMT techniques[C]// Proceedings of the 21st International Conference on Computational Linguistics and 44th Annual Meeting of the Association for Computational Linguistics. 2006: 249-256.
- [39] Yuan Z, Felice M. Constrained grammatical error correction using statistical machine translation[C]// Proceedings of the Seventeenth Conference on Computational Natural Language Learning: Shared Task. 2013: 52-61.
- [40] Chollampatt S, Ng H T. Connecting the dots: Towards human-level grammatical error correction[C]// Proceedings of the 12th Workshop on Innovative Use of NLP for Building Educational Applications. 2017: 327-333.
- [41] Chollampatt S, Taghipour K, Ng H T. Neural network translation models for grammatical error correction[C]// Proceedings of IJCAI. 2016: 2768-2774.
- [42] Sutskever I, Vinyals O, Le Q V. Sequence to sequence learning with neural networks[C]// Proceedings of NIPS. 2014: 3104-3112.
- [43] Gehring J, Auli M, Grangier D, et al. Convolutional sequence to sequence learning[C]// Proceedings of Machine Learning Research: volume 70 Proceedings of ICML. 2017: 1243-1252.
- [44] Bahdanau D, Cho K, Bengio Y. Neural machine translation by jointly learning to align and translate[C]// Proceedings of ICLR. 2015.
- [45] Yuan Z, Briscoe T. Grammatical error correction using neural machine translation[C]// Proceedings of NAACL. 2016: 380-386.
- [46] Chollampatt S, Ng H T. A multilayer convolutional encoder-decoder neural network for grammatical error correction[C]// Proceedings of AAAI. 2018: 5755-5762.
- [47] Grundkiewicz R, Junczys-Dowmunt M, Heafield K. Neural grammatical error correction systems with unsupervised pre-training on synthetic data[C]// Proceedings of the Fourteenth Workshop on Innovative Use of NLP for Building Educational Applications. 2019: 252-263.
- [48] Gu J, Lu Z, Li H, et al. Incorporating copying mechanism in sequence-to-sequence learning[C]// Proceedings of ACL. 2016: 1631-1640.
- [49] Gu J, Bradbury J, Xiong C, et al. Non-autoregressive neural machine translation[C]// Proceedings of ICLR. 2018.
- [50] Liu Y, Ott M, Goyal N, et al. Roberta: A robustly optimized BERT pretraining approach[J]. arXiv preprint arXiv:1907.11692, 2019.
- [51] Yang Z, Dai Z, Yang Y, et al. Xlnet: Generalized autoregressive pretraining for language understanding[C]// Proceedings of NeurIPS 2019. 2019: 5754-5764.
- [52] Awasthi A, Sarawagi S, Goyal R, et al. Parallel iterative edit models for local sequence transduction[C]// Proceedings of EMNLP. 2019: 4260-4270.

-
- [53] Malmi E, Krause S, Rothe S, et al. Encode, tag, realize: High-precision text editing[C]// Proceedings of EMNLP. 2019: 5054-5065.
- [54] Omelianchuk K, Atrasevych V, Chernodub A, et al. GECToR – grammatical error correction: Tag, not rewrite[C]// Proceedings of the Fifteenth Workshop on Innovative Use of NLP for Building Educational Applications. 2020: 163-170.
- [55] Junczys-Dowmunt M, Grundkiewicz R, Guha S, et al. Approaching neural grammatical error correction as a low-resource machine translation task[C]// Proceedings of NAACL. 2018: 595-606.
- [56] Lichtarge J, Alberti C, Kumar S, et al. Corpora generation for grammatical error correction[C]// Proceedings of NAACL. 2019: 3291-3301.
- [57] Hagiwara M, Mita M. GitHub typo corpus: A large-scale multilingual dataset of misspellings and grammatical errors[C]// Proceedings of the 12th Language Resources and Evaluation Conference. 2020: 6761-6768.
- [58] Ge T, Wei F, Zhou M. Fluency boost learning and inference for neural grammatical error correction[C]// Proceedings of ACL. 2018: 1055-1065.
- [59] Wang L, Zhao W, Jia R, et al. Denoising based sequence-to-sequence pre-training for text generation[C]// Proceedings of EMNLP. 2019: 4003-4015.
- [60] Xie Z, Genthial G, Xie S, et al. Noising and denoising natural language: Diverse backtranslation for grammar correction[C]// Proceedings of NAACL. 2018: 619-628.
- [61] Kaneko M, Mita M, Kiyono S, et al. Encoder-decoder models can benefit from pre-trained masked language models in grammatical error correction[C]// Proceedings of ACL. 2020: 4248-4254.
- [62] Radford A, Wu J, Child R, et al. Language models are unsupervised multitask learners[J]. 2019.
- [63] Raffel C, Shazeer N, Roberts A, et al. Exploring the limits of transfer learning with a unified text-to-text transformer[J]. *Journal of Machine Learning Research*, 2020, 21(140): 1-67.
- [64] Tajiri T, Komachi M, Matsumoto Y. Tense and aspect error correction for ESL learners using global context[C]// Proceedings of ACL. 2012: 198-202.
- [65] Yannakoudakis H, Briscoe T, Medlock B. A new dataset and method for automatically grading ESOL texts[C]// Proceedings of ACL. 2011: 180-189.
- [66] Napoles C, Sakaguchi K, Tetreault J. JFLEG: A fluency corpus and benchmark for grammatical error correction[C]// Proceedings of EACL. 2017: 229-234.
- [67] Nicholls D. The cambridge learner corpus: Error coding and analysis for lexicography and elt[C]// Proceedings of the Corpus Linguistics 2003 conference: volume 16. 2003: 572-581.
- [68] Chelba C, Mikolov T, Schuster M, et al. One billion word benchmark for measuring progress in statistical language modeling[J]. arXiv preprint arXiv:1312.3005, 2013.
- [69] Bojar O, Chatterjee R, Federmann C, et al. Findings of the 2017 conference on machine translation (WMT17)[C]// Proceedings of the Second Conference on Machine Translation. 2017: 169-214.
- [70] Wu Y, Schuster M, Chen Z, et al. Google’s neural machine translation system: Bridging the gap between human and machine translation[J]. arXiv preprint arXiv:1609.08144, 2016.

-
- [71] Wan Z, Wan X, Wang W. Improving grammatical error correction with data augmentation by editing latent representation[C]// Proceedings of the 28th International Conference on Computational Linguistics. 2020: 2202-2212.
- [72] Zhang K, Xiong C, Liu Z, et al. Selective weak supervision for neural information retrieval[C]// Proceedings of WWW. 2020: 474-485.
- [73] Williams R J. Simple statistical gradient-following algorithms for connectionist reinforcement learning[J]. Machine learning, 1992, 8(3-4): 229-256.
- [74] Wolf T, Debut L, Sanh V, et al. Transformers: State-of-the-art natural language processing[C]// Proceedings of EMNLP. 2020: 38-45.
- [75] Kingma D P, Ba J. Adam: A method for stochastic optimization[C]// Proceedings of ICLR. 2015.
- [76] Zhang W, Feng Y, Meng F, et al. Bridging the gap between training and inference for neural machine translation[C]// Proceedings of ACL. 2019: 4334-4343.
- [77] Meister C, Vieira T, Cotterell R. Best-first beam search[J]. Transactions of the Association for Computational Linguistics, 2020, 8: 795-809.
- [78] Chollampatt S, Wang W, Ng H T. Cross-sentence grammatical error correction[C]// Proceedings of ACL. 2019: 435-445.
- [79] Kaneko M, Hotate K, Katsumata S, et al. TMU transformer system using BERT for re-ranking at BEA 2019 grammatical error correction on restricted track[C]// Proceedings of the Fourteenth Workshop on Innovative Use of NLP for Building Educational Applications. 2019: 207-212.
- [80] Chollampatt S, Ng H T. Neural quality estimation of grammatical error correction[C]// Proceedings of EMNLP. 2018: 2528-2539.
- [81] Ott M, Auli M, Grangier D, et al. Analyzing uncertainty in neural machine translation[C]// Proceedings of ICML. 2018: 3953-3962.
- [82] Wang S, Liu Y, Wang C, et al. Improving back-translation with uncertainty-based confidence estimation[C]// Proceedings of EMNLP. 2019: 791-802.
- [83] Fomicheva M, Specia L, Guzmán F. Multi-hypothesis machine translation evaluation[C]// Proceedings of ACL. 2020: 1218-1232.
- [84] Hoang D T, Chollampatt S, Ng H T. Exploiting n-best hypotheses to improve an SMT approach to grammatical error correction[C]// Proceedings of IJCAI. 2016: 2803-2809.
- [85] Kreutzer J, Schamoni S, Riezler S. QUality estimation from ScraTCH (QUETCH): Deep learning for word-level translation quality estimation[C]// Proceedings of the Tenth Workshop on Statistical Machine Translation. 2015: 316-322.
- [86] Kim H, Lee J H. A recurrent neural networks approach for estimating the quality of machine translation output[C]// Proceedings of NAACL. 2016: 494-498.
- [87] Kim H, Lee J H, Na S H. Predictor-estimator using multilevel task learning with stack propagation for neural quality estimation[C]// Proceedings of the Second Conference on Machine Translation. 2017: 562-568.
- [88] Başkaya O, Yildiz E, Tunaoğlu D, et al. Integrating meaning into quality evaluation of machine translation[C]// Proceedings of EACL. 2017: 210-219.

-
- [89] Graham Y, Ma Q, Baldwin T, et al. Improving evaluation of document-level machine translation quality estimation[C]// Proceedings of EACL. 2017: 356-361.
- [90] Zhang J, van Genabith J. Translation quality estimation by jointly learning to score and rank[C]// Proceedings of EMNLP. 2020: 2592-2598.
- [91] Ivec J, Blain F, Specia L. deepQuest: A framework for neural-based quality estimation[C]// Proceedings of the 27th International Conference on Computational Linguistics. 2018: 3146-3157.
- [92] Papineni K, Roukos S, Ward T, et al. Bleu: a method for automatic evaluation of machine translation[C]// Proceedings of ACL. 2002: 311-318.
- [93] Peters M, Neumann M, Iyyer M, et al. Deep contextualized word representations[C]// Proceedings of NAACL. 2018: 2227-2237.
- [94] Zhu J, Xia Y, Wu L, et al. Incorporating BERT into neural machine translation[C]// Proceedings of ICLR. 2020.
- [95] Cui Y, Chen Z, Wei S, et al. Attention-over-attention neural networks for reading comprehension[C]// Proceedings of ACL. 2017: 593-602.
- [96] Bryant C, Felice M, Briscoe T. Automatic annotation and evaluation of error types for grammatical error correction[C]// Proceedings of ACL. 2017: 793-805.
- [97] Felice M, Bryant C, Briscoe T. Automatic extraction of learner errors in ESL sentences using linguistically enhanced alignments[C]// Proceedings of COLING. 2016: 825-835.
- [98] Dahlmeier D, Ng H T. Better evaluation for grammatical error correction[C]// Proceedings of NAACL. 2012: 568-572.
- [99] Napoles C, Sakaguchi K, Post M, et al. Ground truth for grammatical error correction metrics[C]// Proceedings of ACL. 2015: 588-593.
- [100] Berthold M R, Höppner F. On clustering time series using euclidean distance and pearson correlation[J]. arXiv preprint arXiv:1601.02213, 2016.
- [101] Clark K, Luong M, Le Q V, et al. ELECTRA: pre-training text encoders as discriminators rather than generators[C]// Proceedings of ICLR. 2020.
- [102] Metzler D, Croft W B. Linear feature-based models for information retrieval[J]. Information Retrieval, 2007.
- [103] Thorne J, Vlachos A, Christodoulopoulos C, et al. FEVER: a large-scale dataset for fact extraction and VERification[C]// Proceedings of NAACL. 2018: 809-819.
- [104] Nie Y, Chen H, Bansal M. Combining fact extraction and verification with neural semantic matching networks[C]// Proceedings of AAAI. 2019: 6859-6866.
- [105] Yoneda T, Mitchell J, Welbl J, et al. UCL machine reading group: Four factor framework for fact finding (HexaF)[C]// Proceedings of the First Workshop on Fact Extraction and VERification (FEVER). 2018: 97-102.
- [106] Chen Q, Zhu X, Ling Z H, et al. Enhanced LSTM for natural language inference[C]// Proceedings of ACL. 2017: 1657-1668.
- [107] Ghaeini R, Hasan S A, Datla V, et al. DR-BiLSTM: Dependent reading bidirectional LSTM for natural language inference[C]// Proceedings of NAACL. 2018: 1460-1469.

-
- [108] Parikh A, Täckström O, Das D, et al. A decomposable attention model for natural language inference[C]// Proceedings of EMNLP. 2016: 2249-2255.
- [109] Radford A, Narasimhan K, Salimans T, et al. Improving language understanding by generative pre-training[C]// Proceedings of Technical report, OpenAI. 2018.
- [110] Li T, Zhu X, Liu Q, et al. Several experiments on investigating pretraining and knowledge-enhanced models for natural language inference[J]. arXiv preprint arXiv:1904.12104, 2019.
- [111] Lin Y, Ji H, Liu Z, et al. Denoising distantly supervised open-domain question answering[C]// Proceedings of ACL. 2018: 1736-1745.
- [112] Xiong C, Dai Z, Callan J, et al. End-to-end neural ad-hoc ranking with kernel pooling[C]// Proceedings of SIGIR. 2017: 55-64.
- [113] Dai Z, Xiong C, Callan J, et al. Convolutional neural networks for soft-matching n-grams in ad-hoc search[C]// Proceedings of WSDM. 2018: 126-134.
- [114] Velickovic P, Cucurull G, Casanova A, et al. Graph attention networks[C]// Proceedings of ICLR. 2018.
- [115] Scarselli F, Gori M, Tsoi A C, et al. The graph neural network model[J]. IEEE Transactions on Neural Networks, 2008: 61-80.
- [116] Kipf T N, Welling M. Semi-supervised classification with graph convolutional networks[C]// Proceedings of ICLR. 2017.
- [117] Luken J, Jiang N, de Marneffe M C. QED: A fact verification system for the FEVER shared task[C]// Proceedings of the First Workshop on Fact Extraction and VERification (FEVER). 2018: 156-160.
- [118] Yin W, Roth D. TwoWingOS: A two-wing optimization strategy for evidential claim verification[C]// Proceedings of EMNLP. 2018: 105-114.
- [119] Guo J, Fan Y, Ai Q, et al. A deep relevance matching model for ad-hoc retrieval[C]// Proceedings of CIKM. 2016: 55-64.
- [120] Hu B, Lu Z, Li H, et al. Convolutional neural network architectures for matching natural language sentences[C]// Proceedings of NIPS. 2014: 2042-2050.
- [121] Pang L, Lan Y, Guo J, et al. Text matching as image recognition[C]// Proceedings of the Thirtieth AAAI. 2016: 2793-2799.
- [122] Dai Z, Callan J. Deeper text understanding for IR with contextual neural language modeling[C]// Proceedings of SIGIR. 2019: 985-988.
- [123] MacAvaney S, Yates A, Cohan A, et al. CEDR: contextualized embeddings for document ranking[C]// Proceedings of SIGIR. 2019: 1101-1104.
- [124] Knyazev B, Taylor G W, Amer M R. Understanding attention and generalization in graph neural networks[C]// Proceedings of NeurIPS. 2019: 4204-4214.
- [125] Fleiss J L. Measuring nominal scale agreement among many raters.[J]. Psychological bulletin, 1971, 76(5): 378.
- [126] Nie Y, Wang S, Bansal M. Revealing the importance of semantic retrieval for machine reading at scale[C]// Proceedings of EMNLP. 2019: 2553-2566.

-
- [127] Gardner M, Grus J, Neumann M, et al. AllenNLP: A deep semantic natural language processing platform[C]// Proceedings of Workshop for NLP Open Source Software (NLP-OSS). 2018: 1-6.
- [128] Nogueira R, Cho K. Passage re-ranking with bert[J]. arXiv preprint arXiv:1901.04085, 2019.
- [129] Guo J, Fan Y, Pang L, et al. A deep look into neural ranking models for information retrieval[J]. Information Processing & Management, 2020, 57(6): 102067.
- [130] Liu T. Learning to rank for information retrieval[C]// Proceeding of SIGIR. 2010: 904.
- [131] Zhao C, Xiong C, Rosset C, et al. Transformer-xh: Multi-evidence reasoning with extra hop attention[C]// Proceedings of ICLR. 2020.
- [132] Ye D, Lin Y, Du J, et al. Coreferential Reasoning Learning for Language Representation[C]// Proceedings of EMNLP. 2020: 7170-7186.
- [133] Clark K, Khandelwal U, Levy O, et al. What does BERT look at? an analysis of BERT's attention[C]// Proceedings of the 2019 ACL Workshop BlackboxNLP: Analyzing and Interpreting Neural Networks for NLP. 2019: 276-286.
- [134] Cinelli M, Quattrocioni W, Galeazzi A, et al. The covid-19 social media infodemic[J]. arXiv preprint arXiv:2003.05004, 2020.
- [135] Hossain T, Logan IV R L, Ugarte A, et al. Detecting covid-19 misinformation on social media[J]. 2020.
- [136] Li Y, Grandison T, Silveyra P, et al. Jennifer for COVID-19: An NLP-powered chatbot built for the people and by the people to combat misinformation[C]// Proceedings of the 1st Workshop on NLP for COVID-19 at ACL 2020. 2020.
- [137] Medina Serrano J C, Papakyriakopoulos O, Hegelich S. NLP-based feature extraction for the detection of COVID-19 misinformation videos on YouTube[C]// Proceedings of the 1st Workshop on NLP for COVID-19 at ACL 2020. 2020.
- [138] Craswell N, Mitra B, Yilmaz E, et al. Overview of the trec 2019 deep learning track[J]. arXiv preprint arXiv:2003.07820, 2020.
- [139] Zhang H, Song X, Xiong C, et al. Generic intent representation in web search[C]// Proceedings of SIGIR. 2019: 65-74.
- [140] Sennrich R, Haddow B, Birch A. Neural machine translation of rare words with subword units[C]// Proceedings of ACL. 2016: 1715-1725.
- [141] Kudo T, Richardson J. SentencePiece: A simple and language independent subword tokenizer and detokenizer for neural text processing[C]// Proceedings of EMNLP. 2018: 66-71.
- [142] Bajaj P, Campos D, Craswell N, et al. Ms marco: A human generated machine reading comprehension dataset[J]. arXiv preprint arXiv:1611.09268, 2016.
- [143] Roberts K, Alam T, Bedrick S, et al. Trec-covid: Rationale and structure of an information retrieval shared task for covid-19[J]. Journal of the American Medical Informatics Association, 2020.
- [144] Yang W, Zhang H, Lin J. Simple applications of bert for ad hoc document retrieval[J]. arXiv preprint arXiv:1903.10972, 2019.
- [145] Beltagy I, Cohan A, Lo K. Scibert: Pretrained contextualized embeddings for scientific text[J]. arXiv preprint arXiv:1903.10676, 2019.

-
- [146] Lee J, Yoon W, Kim S, et al. Biobert: a pre-trained biomedical language representation model for biomedical text mining[J]. *Bioinformatics*, 2020, 36(4): 1234-1240.
- [147] Wang L L, Lo K, Chandrasekhar Y, et al. CORD-19: The COVID-19 open research dataset[C]// *Proceedings of the 1st Workshop on NLP for COVID-19 at ACL 2020*. 2020.
- [148] Voorhees E, Alam T, Bedrick S, et al. Trec-covid: Constructing a pandemic information retrieval test collection[J]. *arXiv preprint arXiv:2005.04474*, 2020.
- [149] Sun S, Qian Y, Liu Z, et al. Meta adaptive neural ranking with contrastive synthetic supervision[J]. *arXiv preprint arXiv:2012.14862*, 2020.
- [150] Gururangan S, Marasović A, Swayamdipta S, et al. Don't stop pretraining: Adapt language models to domains and tasks[C]// *Proceedings of ACL*. 2020: 8342-8360.
- [151] Lo K, Wang L L, Neumann M, et al. S2ORC: The semantic scholar open research corpus[C]// *Proceedings of ACL*. 2020: 4969-4983.
- [152] MacAvaney S, Cohan A, Goharian N. Sledge: A simple yet effective baseline for coronavirus scientific knowledge search[J]. *arXiv preprint arXiv:2005.02365*, 2020.
- [153] Xiong L, Xiong C, Li Y, et al. Approximate nearest neighbor negative contrastive learning for dense text retrieval[J]. *arXiv preprint arXiv:2007.00808*, 2020.
- [154] Luan Y, Eisenstein J, Toutanova K, et al. Sparse, dense, and attentional representations for text retrieval[J]. *arXiv preprint arXiv:2005.00181*, 2020.
- [155] Karpukhin V, Oguz B, Min S, et al. Dense passage retrieval for open-domain question answering[C]// *Proceedings of EMNLP*. 2020: 6769-6781.

致 谢

衷心感谢我的导师孙茂松老师对我的悉心指导以及在攻读博士的五年时间里对我的包容和关怀。感谢孙茂松老师为我提供的良好的科研环境以及各种各样的学术交流机会。他严谨的治学态度、实事求是的科研精神对我产生了深刻的影响。十分幸运能成为孙老师的学生。

衷心感谢刘知远老师在我学术和工作上的意见和建议，刘老师的科研态度以及学术视野成为了我科研道路上的榜样。

衷心感谢刘洋老师的言传身教，他追求卓越和严谨踏实的研究态度，时刻感染着我。我希望自己能够像他一样严于律己。

感谢熊辰炎师兄在我读博期间对我的帮助，他的严谨的治学态度以及学术思想，都让我受益匪浅。

感谢 THUNLP 实验室的每一名同学在我攻读博士学位的五年时间中给予我的关怀与帮助，是你们的陪伴与鼓励让我能够在读博期间能够克服困难。

最后，感谢我的父亲刘恒哲先生，母亲臧守艳女士以及我的家人们在我读博期间给予我的包容和支持，让我能够坚持追求自己的目标。

声 明

本人郑重声明：所呈交的学位论文，是本人在导师指导下，独立进行研究工作所取得的成果。尽我所知，除文中已经注明引用的内容外，本学位论文的研究成果不包含任何他人享有著作权的内容。对本论文所涉及的研究工作做出贡献的其他个人和集体，均已在文中以明确方式标明。

签 名：_____ 日 期：_____

个人简历、在学期间完成的相关学术成果

个人简历

1994年10月22日出生于辽宁省本溪市。

2012年9月考入东北大学计算机科学与技术系计算机科学与技术专业，2016年7月本科毕业并获得工学学士学位。

2016年9月免试进入清华大学计算机科学与技术系攻读工学博士至今。

在学期间完成的相关学术成果

学术论文：

- [1] **Zhenghao Liu**, Chenyan Xiong, Maosong Sun, Zhiyuan Liu. Fine-grained Fact Verification with Kernel Graph Attention Network. The Annual Meeting of the Association for Computational Linguistics (ACL 2020). (CCF A).
- [2] **Zhenghao Liu**, Chenyan Xiong, Maosong Sun, Zhiyuan Liu. Adapting Open Domain Fact Extraction and Verification to COVID-FACT through In-Domain Language Modeling. The Conference on Empirical Methods in Natural Language Processing (EMNLP 2020: Findings). (CCF B).
- [3] **Zhenghao Liu**, Xiaoyuan Yi, Maosong Sun, Liner Yang, Tat-Seng Chua. Neural Quality Estimation with Multiple Hypotheses for Grammatical Error Correction. The 2021 Conference of the North American Chapter of the Association for Computational Linguistics - Human Language Technologies (NAACL-HLT 2021). (CCF C).
- [4] **Zhenghao Liu***, Kaitao Zhang*, Chenyan Xiong, Zhiyuan Liu, Maosong Sun. OpenMatch: An Open Source Library for Neu-IR Research. The 44th International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR 2021). (CCF A). (* 代表同等贡献).
- [5] Chenyan Xiong*, **Zhenghao Liu***, Si Sun*, Zhuyun Dai*, Kaitao Zhang*, Shi Yu*, Zhiyuan Liu, Hoifung Poon, Jianfeng Gao, Paul Bennett. CMT in TREC-COVID Round 2: Mitigating the Generalization Gaps from Web to Special Domain Search. 2020. (* 代表同等贡献).
- [6] **Zhenghao Liu**, Chenyan Xiong, Maosong Sun, Zhiyuan Liu. Entity-Duet Neural Ranking: Understanding the Role of Knowledge Graph Semantics in Neural Information Retrieval. The Annual Meeting of the Association for Computational Linguistics (ACL 2018). (CCF A).

- [7] **Zhenghao Liu**, Chenyan Xiong, Maosong Sun, Zhiyuan Liu. Explore Entity Embedding Effectiveness in Entity Retrieval. China National Conference on Chinese Computational Linguistics (CCL 2019). (**EI**).
- [8] Houyu Zhang*, **Zhenghao Liu***, Chenyan Xiong, Zhiyuan Liu. Grounded Conversation Generation as Guided Traverses in Commonsense Knowledge Graphs. The Annual Meeting of the Association for Computational Linguistics (ACL 2020). (**CCF A**). (* 代表同等贡献).
- [9] Shi Yu*, **Zhenghao Liu***, Chenyan Xiong, Tao Feng, Zhiyuan Liu. Few-Shot Conversational Dense Retrieval. The 44th International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR 2021). (**CCF A**). (* 代表同等贡献).
- [10] Xiaoyuan Yi, **Zhenghao Liu**, Wenhao Li, Maosong Sun. Text Style Transfer via Learning Style Instance Supported Latent Space. The International Joint Conference on Artificial Intelligence (IJCAI 2020). (**CCF A**).
- [11] Si Sun, Yingzhuo Qian, **Zhenghao Liu**, Chenyan Xiong, Kaitao Zhang, Jie Bao, Zhiyuan Liu, Paul Bennett. Few-Shot Text Ranking with Meta Adapted Synthetic Weak Supervision. The Joint Conference of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (ACL-IJCNLP 2021). (**CCF A**).
- [12] Kaitao Zhang, Chenyan Xiong, **Zhenghao Liu**, Zhiyuan Liu. Selective Weak Supervision for Neural Information Retrieval. The Web Conference (WebConf 2020). (**CCF A**).
- [13] Deming Ye, Yankai Lin, Jiaju Du, **Zhenghao Liu**, Maosong Sun, Zhiyuan Liu. Coreferential Reasoning Learning for Language Representation. The Conference on Empirical Methods in Natural Language Processing (EMNLP 2020). (**CCF B**).
- [14] Liner Yang, Maosong Sun, Jiacheng Zhang, **Zhenghao Liu**, Huanbo Luan, Yang Liu. Neural Parse Combination. Journal of Computer Science and Technology (JCST 2017). (**CCF B**).
- [15] Yuan Yao, Deming Ye, Peng Li, Xu Han, Yankai Lin, **Zhenghao Liu**, Zhiyuan Liu, Lixin Huang, Jie Zhou, Maosong Sun. DocRED: A Large-Scale Document-Level Relation Extraction Dataset. The Annual Meeting of the Association for Computational Linguistics (ACL 2019). (**CCF A**).
- [16] 梁健楠, 孙茂松, 矣晓沅, 杨成, 陈慧敏, 刘正皓. 基于神经网络的集句诗自动生成. 中文信息学报. (**CCL 2018 最佳论文**).

指导教师学术评语

文本校对是自然语言处理一个重要研究与应用方向。本论文着眼于文本校对中语法纠错和事实验证校对两个任务，从多个角度开展研究。论文选题具有重要的理论意义和应用价值。

在对国内外研究现状充分调研和了解的基础上，作者独立完成的主要工作和创新性成果如下：

1、针对英文文本校对中的语法纠错任务，通过对比通用预训练语言模型以及面向语法纠错的预训练策略，对基于生成的语法纠错模型进行了优化，取得了令人满意的实验结果。同时针对于语法纠错语料库中存在噪音的问题，提出了一种基于强化学习的数据筛选方法，提升了语法纠错的效果。

2、进一步提出了一种融合多证据的英文语法纠错方法，通过所设计的两种注意力机制在多个语法纠错结果中传播语法纠错证据，提升预训练语言模型在语法错误检查以及语法纠错质量评估任务上的效果，并通过对语法纠错结果重排序进一步提升语法纠错效果。

3、针对英文文本校对中的事实验证任务，提出了一种融合多事实证据的细粒度联合推理的事实验证方法。通过建立基于多事实证据的推理图及带有核函数的注意力机制，实现了多证据之间的细粒度联合推理。

4、进一步提出了一种面向专业领域的事实验证算法，通过强化预训练语言模型对专业领域文本的理解能力，提升专业领域事实验证的效果。

论文结构合理，条理清晰，叙述清楚，写作规范，达到了工学博士学位论文水平，是一篇优秀的博士论文。

答辩委员会决议书

文本校对是自然语言处理领域的重要研究方向之一。论文针对语法检查和事实验证展开研究工作，选题具有重要的学术意义和应用价值。

论文的主要工作和创新性成果如下：

(1) 提出了一种基于强化学习的数据筛选方法，以及针对语法改错的预训练语料增广策略，有效提升了语法改错效果。

(2) 提出了一种融合多证据的语法改错质量评估模型，通过对语法改错结果重排序，有效提升了语法改错效果。

(3) 提出了一种基于多证据联合推理的事实验证模型，并基于核函数的注意力机制实现了细粒度的联合推理，实验验证了模型的有效性。

(4) 提出了一种面向专业领域的事实验证方法，通过专业领域的预训练语言模型，显著提升了事实验证效果。

论文工作表明，该同学已经掌握了本学科坚实宽广的基础理论和系统深入的专门知识，独立从事科学研究工作的能力强。论文表述清楚，逻辑合理，写作规范，达到了工学博士学位论文的水平，是一篇优秀的博士论文。答辩过程中叙述清楚，回答问题正确。答辩委员会经无记名投票，一致同意刘正皓同学通过博士学位论文答辩，并建议授予工学博士学位。