学校代码：10284
分 类 号：TP181
密　　级：公开
U D C：004.8
学　　号：MG21330048

南京大学

硕 士 学 位 论 文

论 文 题 目　基于数据增强与上下文学习的命名实体识别研究

作 者 姓 名　宋斯涵

专 业 名 称　计算机科学与技术

研 究 方 向　自然语言处理

导 师 姓 名　申富饶 教授

2024 年 5 月 27 日

答辩委员会主席 戴新宇 教授

评　　阅　　人 戴新宇 教授

徐明华 教授

论文答辩日期　2024 年 5 月 16 日

研究生签名:

导师签名:

Research on Named Entity Recognition based on Data Augmentation and In-Context Learning

by

Song Sihan

Supervised by

Professor Shen Fu-Rao

A dissertation submitted to

the graduate school of Nanjing University

in partial fulfilment of the requirements for the degree of

MASTER

in

Computer Science and Technology

Department of Computer Science and Technology

Nanjing University

May 27, 2024

南京大学研究生毕业论文中文摘要首页用纸

毕业论文题目：基于数据增强与上下文学习的命名实体识别研究

计算机科学与技术 专业 2021 级硕士生姓名：宋斯涵

指导教师（姓名、职称）：申富饶 教授

摘　要

命名实体识别是自然语言处理中的一项基础任务，在信息抽取、知识图谱构建和机器翻译等多个下游任务中发挥着重要作用。然而，命名实体识别在实际应用中面临着诸多挑战，严重限制了该任务的性能表现。最大的挑战来自于标注数据的稀缺性。数据是深度学习的基础，本文首先聚焦于数据层面，研究如何为该任务生成高质量和高多样性的增强数据，从而为后续的工作奠定基础。随着大模型的兴起，上下文学习已经成为自然语言处理领域的流行范式。在大模型时代，命名实体识别面临的另一大挑战是现有的上下文学习算法在该任务上的表现不佳，没有充分发挥出大模型的潜力。本文进一步着眼于大模型的应用，致力于为该任务设计优秀的上下文学习算法。本文围绕上述两大挑战进行研究，完成了以下工作：

1. 本文从数据的角度出发，提出了一种基于提示的数据增强算法，简称RoPDA。该算法在生成模型中加入了连续提示，并通过仅更新提示向量的参数来适应下游的数据增强任务。为了提高增强样本的多样性，RoPDA 通过多种基本的增强操作来同时增强实体和上下文，并生成标签翻转和标签保留的增强样本。考虑到增强样本中存在一定的噪声，RoPDA 借助模型的自一致性来过滤掉低质量的增强样本。经实验验证，该算法能够为命名实体识别任务产生高质量和高多样性的增强样本，从而为后续研究奠定良好的数据基础。

2. 为了充分发挥大模型在命名实体识别任务上的潜力，本文提出了一种基于思维链与示例选择的上下文学习算法，简称 CoTIS-NER。该算法明确地将命名实体识别任务分解为三个连续的子问题后进行多步推理，并通过引入负样本推理信息来提高实体预测的全面性和准确性。为了帮助测试示例选择到合适的演示示例，CoTIS-NER 首先使用 RoPDA 算法来进行样本集扩充，随后针对该任

务的特点设计了一种同时考虑句子语义信息和实体信息的示例选择策略。实验结果表明该算法显著提升了大模型在命名实体识别任务上的表现。

3. 基于提出的两个算法，本文设计并开发了一个命名实体识别系统。该系统可以满足用户在多个应用领域的命名实体识别需求，并且具备通用性与实时性，充分展示了本文研究内容的实际应用价值。

关键词：命名实体识别；自然语言处理；数据增强；上下文学习

南京大学研究生毕业论文英文摘要首页用纸

THESIS: Research on Named Entity Recognition based on Data Augmentation and In-Context Learning

SPECIALIZATION: Computer Science and Technology

POSTGRADUATE: Song Sihan

MENTOR: Professor Shen Fu-Rao

ABSTRACT

Named entity recognition is a fundamental task in natural language processing and plays an important role in many downstream tasks such as information extraction, knowledge graph construction and machine translation. However, named entity recognition faces many challenges in practical applications, which severely limits its performance. One of the biggest challenges comes from the scarcity of annotated data. Data is the foundation of deep learning. Therefore, this paper first focuses on the data level and studies how to generate high quality and diverse augmented data for named entity recognition tasks, laying the foundation for subsequent work. With the rise of large language models, in-context learning has become a popular paradigm in the field of natural language processing. In the era of large language models, another major challenge facing named entity recognition is that existing in-context learning methods perform poorly on it and do not fully utilize the potential of large language models. This paper further focuses on the application of large language models, aiming at designing an excellent in-context learning algorithm for named entity recognition tasks. This paper focuses on the above two challenges and has completed the following work:

1. From the perspective of data, this paper proposes a prompt-based data augmentation algorithm, referred to as RoPDA. RoPDA adds continuous prompt to the generation model and learns to adapt to downstream data augmentation tasks by only updating the parameters of prompt vectors. To improve the diversity of augmented samples, RoPDA performs entity augmentation and context augmentation through multiple fundamental augmentation operations to generate label-flipping and label-preserving sam-

ples. Considering the noise in the augmented samples, RoPDA filters out low-quality augmented samples with the help of the self-consistency of the model. Experimental results show that RoPDA can generate high-quality and high-diversity augmented samples for named entity recognition, thus laying a good data foundation for subsequent research.

2. In order to fully exploit the potential of large language models, this paper proposes an in-context learning algorithm based on chain-of-thought and instance selection for named entity recognition, referred to as CoTIS-NER. CoTIS-NER explicitly decomposes the named entity recognition task into three consecutive sub-problems for multi-step reasoning and improves the comprehensiveness and accuracy of entity prediction by introducing reasoning information from negative samples. In order to select good demonstration examples for test instances, CoTIS-NER first leverages RoPDA to expand the sample set, and then designs an instance selection strategy that considers both sentence semantic information and entity information based on the characteristics of named entity recognition. Experimental results show that CoTIS-NER significantly improves the performance of large language models on named entity recognition.

3. This paper designs and develops a named entity recognition system based on the two proposed algorithms. The system can meet the user's needs for named entity recognition in many application fields, which is universal and real-time, embodying the practical application value of our research.

KEYWORDS: Named Entity Recognition; Natural Language Processing; Data Augmentation; In-Context Learning

目　　录

插图目录

表格目录

第一章　绪论

1.1　研究背景与意义

随着互联网和数字化技术的快速发展，海量的文本数据被产生和存储，这些文本数据中蕴含着丰富的信息，如何运用自然语言处理（Natural Language Processing，NLP）技术来深入挖掘和理解其中的关键信息变得至关重要。命名实体识别（Named Entity Recognition，NER）是自然语言处理中的一项基础任务，其目标是从文本中自动识别出具有特定意义的命名实体，如人名、地名、组织机构名和日期等。这些命名实体在文本中承载着重要的语义信息，对于理解和分析文本内容具有重要意义。命名实体识别在许多下游任务上都有着广泛的应用。在对话系统中，命名实体识别能够将用户隐式的意图转化为显式的指令，从而让计算机理解并提供准确的回答；在信息检索中，命名实体识别可以快速定位和检索文本中的关键词，从而更好地理解用户的查询意图；在知识图谱构建中，命名实体识别可以自动抽取文本中的实体信息，从而构建丰富的知识图谱。

命名实体识别作为自然语言处理领域的关键任务，已经在金融、医疗和法律等多个领域显示出其应用价值。然而，这一任务在实际应用中面临着一系列挑战，其中最为显著的挑战之一是标注数据的稀缺性。数据是深度学习的基础，深度学习需要大量的标注数据来训练模型，因此数据的规模和质量是影响深度学习模型性能的关键因素。在 NER 的实际应用中，尤其是在医疗和法律等高度专业化的领域，由于数据采集困难以及标注成本高昂，获取足够数量的标注数据以支持模型训练变得十分困难。在数据量有限的情况下，训练出的 NER 模型难以学习到准确的特征和模式，极易陷入过拟合的问题，从而使得模型在真实场景中性能很差。在这一背景下，研究适用于命名实体识别任务的数据增强技术从而提升标注数据的数量和多样性，可以增强模型在样本稀缺场景下的泛化能力和准确性，对于实际应用具有重要的现实意义。这一研究能够降低数据标

注的成本和人力投入，使得 NER 更好地适应不同领域和场景的需求，为广泛应用 NER 提供了更为经济和可行的解决方案。

在深度学习中，足够数量的标注数据是训练出高质量命名实体识别模型的关键基础。然而随着算力资源的大幅增加，我们可以借助更为复杂和强大的通用型模型来完成各种下游任务。ChatGPT[1]等大型语言模型（Large Language Models，LLM，又可称作大模型）的推出，标志着我们正式进入了大模型时代。通过在海量文本数据上进行训练，大模型积累了广泛的通用知识，具有理解与生成人类语言、提供信息和进行对话互动的非凡能力，在各个下游任务和应用领域上展现出了巨大的潜力。在命名实体识别任务中，大模型强大的语言理解和推理能力使其能够更好地理解文本中的上下文和复杂语义结构，并推断出实体在文本中的角色，从而更加准确地理解和识别实体。此外，大模型通过学习海量数据所获得的世界知识也有助于更好地理解文本的背景信息，对于准确判断实体类型具有重要作用。因此，使用大模型来完成 NER 任务有着巨大的潜力，是一个值得探索的方向。

大模型展现出的通用智能与其上下文学习（In-Context Learning，ICL）的能力密切相关。上下文学习是一种全新的自然语言处理范式，它无需对模型参数进行调整，通过引入少量示例来帮助大模型更好地理解语境并从中学习，在不同的下游任务中展现出了强大的通用性和适应性。尽管如此，在 NER 任务中，上下文学习的应用尚未达到预期效果，面临着一系列亟待解决的问题，包括对示例组织高度敏感、缺乏有效的提示策略设计和未能充分利用大模型的推理能力等。这些问题严重限制了 NER 任务的性能表现，未能充分发挥出大模型的潜力，是该任务在大模型时代面临的一大挑战。因此，研究如何提升上下文学习在命名实体识别任务上的表现显得尤为重要，这不仅为命名实体识别提供了更为高效和灵活的解决途径，也有助于进一步拓展大模型的应用领域。

在深度学习蓬勃发展的背景下，命名实体识别仍然存在着两大亟待解决的挑战：标注数据的稀缺以及上下文学习在该任务上的表现不佳。本文将围绕这两大挑战进行研究，首先从数据的角度出发，为该任务生成大量高质量、高多样性的增强样本，从而为后续的研究奠定基础；随后则站在大模型的肩膀上，为该任务探索 training-free 的最佳上下文学习方案。本文将深入探讨和分析这些方法的原理和实验结果，以期促进命名实体识别的研究和发展。

1.2 研究现状与问题

针对上述两大挑战，本节深入分析了数据增强和上下文学习在命名实体识别中的研究现状以及存在的问题。

1.2.1 数据增强

数据增强指通过对已有数据添加微小改动或者从已有数据合成新数据，从而达到扩充数据集的目的[2]。数据增强可以缓解深度学习中的数据稀缺问题，从而提高模型的泛化能力和鲁棒性，减少过拟合的风险。通过数据增强来提升命名实体识别任务的性能是近年来的研究热点之一[3-6]。在命名实体识别任务中，常见的数据增强方法大致可以分为四类：基于规则的方法、基于样本插值的方法、基于预训练语言模型的方法以及反向翻译。

基于规则的方法是一类最常见的数据增强手段，通过使用预定义的规则来对原始数据进行变换。Wei 等人[7]通过同义词替换、随机插入、随机删除和随机交换等多种规则来对原始文本进行随机扰动，从而生成新样本。尽管这种方法简单易实现，但可能会破坏句子的完整语法和句法结构并造成单词与实体标签不一致的现象。Dai 等人[5]提出将文本中的实体随机替换为训练集中同一类型的其他实体，从而避免了单词与实体标签不一致的问题，但是在该方法中，实体的多样性并没有增加，并且替换后的实体与上下文可能并不匹配。

部分研究者[8-10]提出插值类方法，通过对两个样本在输入层或者中间表示层进行 Mixup[11]来生成增强样本。Mixup 是一种在输入空间中对模型进行平滑的正则化方法，有助于模型学习到更加平滑的决策边界，从而防止过拟合。Zhang 等人[8]研究了全序列混合、子序列混合以及标签约束混合对于序列标注任务的影响。但是直接进行 Mixup 会引入过多噪声，在序列标注任务中可能会给模型训练带来负面影响。为了使得模型更好地泛化和收敛，CIAug[9]将 Mixup 与课程学习[12]相结合，在训练过程中根据空间距离逐渐增加 Mixup 样本的难度。LADA[10]提出通过 kNN 来选取距离相近的文本 x'，并将原始文本 x 与 x' 进行 Mixup，从而减少噪声。

近年来，由于预训练语言模型（Pre-trained Language Models，PLM）具备强大的生成能力以及丰富的通用知识，研究者们开始探索如何借助预训练语言模

型来生成增强数据。使用预训练语言模型的优点在于生成的句子质量好，流畅性高。Zhou 等人[4]提出对原始文本中的实体进行随机掩码后，使用掩码语言模型 BERT[13]来生成新的实体，另外，他们通过将实体标签显式注入到文本上下文中来缓解单词和实体标签不一致的问题。这种方法的局限性在于仅提升了实体多样性，并没有增加上下文多样性，从而导致增强数据的多样性不足。Wang 等人[3]和 Anaby-Tavor 等人[14]利用生成式预训练语言模型以自回归的方式来生成增强数据。但这类方法通常需要借助外界语料库来进行额外的预训练。

此外，还有一类方法借助机器翻译，通过将源语言文本翻译为目标语言，再将目标语言文本翻译回源语言，来产生新的样本，这类方法称为反向翻译[15]。反向翻译试图通过引入语法和句法的变化来增加数据的多样性。但是在命名实体识别任务中，反向翻译引入的数据多样性终究是十分有限的，并且难以准确定位增强文本中的实体位置，从而给文本标签的准确标注带来较多噪声。

尽管自然语言处理领域的数据增强方法已经相当成熟，但大部分方法仅适用于句子级别的任务，命名实体识别则属于单词级别的任务。在命名实体识别任务中，某个单词的变化不仅会导致自身标签发生变化，还可能影响上下文中其他单词的标签，这一特性使得许多句子级数据增强方法在命名实体识别任务中难以达到预期效果。尽管上述方法大多是专门为命名实体识别任务设计的，但这些增强方法仍然存在着一些问题，比如破坏句子结构、单词和实体标签不匹配以及依赖于外部知识库或语料库等。因此，探索适用于命名实体识别任务的数据增强方法具有重要的研究价值。

1.2.2 上下文学习

随着大型语言模型能力的不断提升，上下文学习已经成为一种新兴的自然语言处理范式，它通过将相关指令和演示示例拼接到提示中来引导模型进行预测，在许多 NLP 任务上取得了令人印象深刻的表现[16]，展现出了极强的通用性和泛化性。然而目前上下文学习仍然面临着一些挑战，如表现不稳定，对提示格式和示例组织等因素十分敏感[17-19]，以及幻觉[20-21]问题导致输出不准确等。这些挑战导致大模型在命名实体识别任务上的性能表现有待提升。

一些研究工作通过设计有效的提示格式来提升上下文学习在 NER 任务上的表现。Wang 等人[22]认为大模型在 NER 任务上效果不佳的根本原因是：NER 是

序列标注任务，而 LLM 则更擅长文本生成任务，这两种任务之间存在着天然的差异。他们通过在提示中引导大模型使用特殊标记包围实体来生成标记序列，从而将 NER 任务转化为大模型擅长的文本生成任务。Wei 等人[23]通过问题分解，将 NER 任务转化为一个两阶段框架的多轮问答问题：首先过滤出文本中可能存在的实体类型，以减少搜索空间和计算复杂度，随后依次对每种可能的类型进行链式信息抽取。为了实现对实体的准确判断，执行 NER 任务不仅需要利用 LLM 的文本理解能力和海量世界知识，还需要发挥一定的推理能力。Xie 等人[24]尝试将多种流行的推理技术运用于 NER 任务：将该任务按照标签分解为一系列子问题；利用句法激励和工具增强来激发中间思维；通过两阶段多数投票策略来将自一致性应用到 NER 任务中。为了进一步释放大模型的推理潜力，一种可行的方法是将复杂问题的求解分解为一系列的中间推理步骤，即思维链[25]。最近思维链被用于多种 NLP 下游任务并取得了优异的表现。Wang 等人[26]提出了摘要思维链，旨在引导大模型逐步生成摘要。Ma 等人[27]利用思维链来进行关系抽取，引导大模型生成有助于理解实体关系的中间证据。本文从他们的工作中受到启发，研究将思维链技术应用于 NER 任务上，通过生成显式中间思考过程来辅助大模型更准确地判断实体类型。

研究表明，上下文学习的性能不仅依赖于提示的格式，还依赖于演示示例的选择和排列顺序[17-19]。Liu 等人[19]发现选择最近邻作为示例是一个很好的解决方案，他们提出了 KATE，首先计算文本向量之间的距离（如欧式距离），随后通过 kNN 来选取示例。Levy 等人[28]则认为仅选择相似的示例是不够的，他们通过选择多样化的示例来提供更多有用的信息并提高组合泛化能力。还有一些方法利用大模型的输出分数作为指标来选择演示示例[29-31]。近来关注上下文示例选择的研究工作有很多，然而现有的示例选择方法大多针对句子级任务，使用句子的整体语义来进行示例选择，而 NER 则是一个更加关注局部关联的单词级任务。为了解决这一问题，Wang 等人[22]提出了一种单词级的最近邻示例检索策略，但是该策略中需要微调一个 NER 分类模型。Wan 等人[32]提出通过融合实体对信息来重构上下文，从而在检索过程中既保留了句子的语义信息，又保留了以实体对为中心的信息，但是这种方法仅适用于实体信息已经确定的关系抽取任务。迄今为止，尚未有研究致力于设计符合 NER 任务特性且无需训练的示例选择策略，这是提升上下文学习在该任务上的性能的一大挑战。

1.3 研究内容与贡献

本文围绕命名实体识别任务进行研究，首先从数据的角度出发，研究如何利用预训练语言模型来为该任务生成质量高、多样性强的新样本以解决标注数据稀缺的挑战，并提出了一种全新的数据增强算法，从而为后续的工作奠定数据基础。其次以大模型为着手点，探索了如何更好地利用大型语言模型的通用知识和推理能力来进行命名实体识别，在此基础上提出了一种有效的上下文学习算法。随后，本文将这两种算法应用于领域通用的命名实体识别系统中，以验证算法的有效性。本文的主要研究内容与贡献如下：

- 本文从数据的角度出发，提出了一种基于连续提示的数据增强算法 RoPDA。为了使得模型在小样本下也能充分学习并适应下游的数据增强任务，RoPDA 在预训练语言模型中加入了连续提示，并且在训练过程中仅对提示向量的参数进行更新。为了提高增强样本的多样性，RoPDA 通过五种基本增强操作来进行实体增强和上下文增强，以生成标签翻转和标签保留的增强样本。考虑到增强样本中存在一定的噪声，RoPDA 设计了一种基于双向掩码的微调方式来使模型具备自一致过滤的能力，以过滤掉低质量的增强样本。多个基准数据集上的实验结果表明，RoPDA 的表现显著优于目前最先进的数据增强算法，能够为命名实体识别任务生成高质量和高多样性的增强样本。
- 本文针对上下文学习在命名实体识别任务上表现不佳的挑战，提出了一种基于思维链与示例选择的上下文学习算法 CoTIS-NER。该算法明确地将命名实体识别任务分解为三个连续的子问题并进行多步推理，在此过程中通过引入负样本推理信息来提高实体预测的全面性和准确性。CoTIS-NER 使用真实标签来引导大模型自动为支持集样本生成候选实体和显式思考过程。为了帮助问题查询选择到更加合适的演示示例，CoTIS-NER 首先使用 RoPDA 算法来对标注样本集进行扩充，随后针对命名实体识别任务的特点设计了一种融合实体信息的示例选择策略。多个基准数据集和多个不同规模大模型上的实验结果显示，CoTIS-NER 显著提升了大模型在命名实体识别任务上的性能。
- 本文设计了一个领域通用的命名实体识别系统，并集成了本文提出的两个命名实体识别算法。该系统为用户提供了一个简洁直观的操作界面，使得

用户能够轻松地对多个领域的文本数据执行命名实体识别任务，并迅速获得准确可靠的识别结果，充分体现了本研究所提出的算法在实际应用中的高效性和实用性。

1.4 论文纲要

本文由六个章节构成，文章的总体结构如图1-1所示，各章的主要内容概述如下：

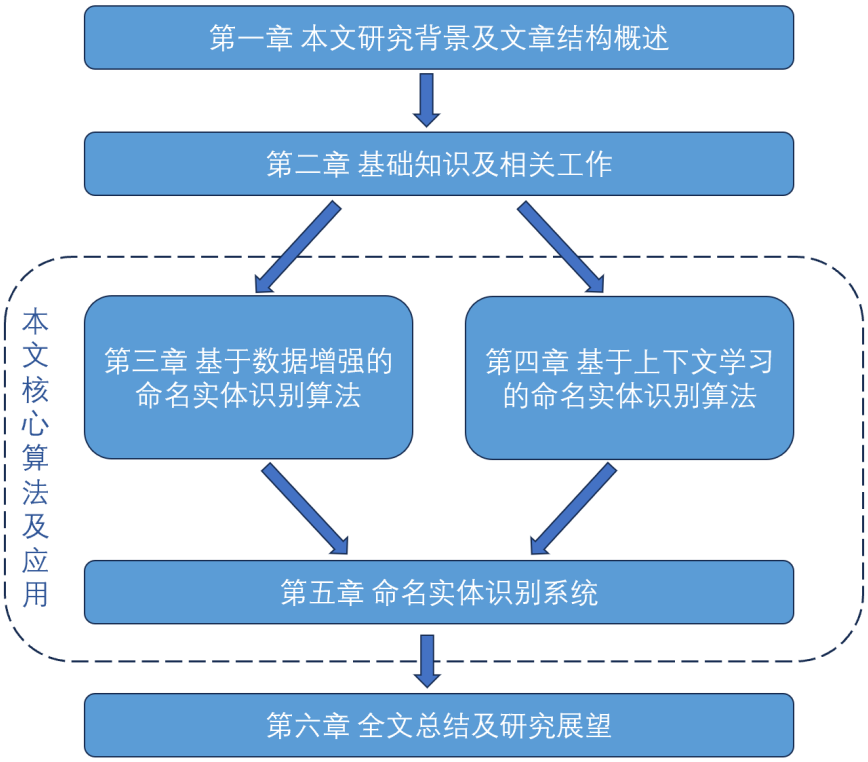


图 1-1 本文整体结构

第一章为绪论，主要介绍了命名实体识别任务的相关背景、研究进展与研究挑战，以及本文针对这些挑战所进行的研究内容及主要贡献。

第二章为相关工作，首先给出命名实体识别的任务介绍与基本模型，随后对本文中涉及的预训练语言模型、提示学习和上下文学习等概念进行介绍。

第三章介绍了本文提出的基于连续提示的数据增强算法 RoPDA，首先由当前数据增强算法存在的问题引出研究动机；然后详细阐述了 RoPDA 的设计细节，以及如何借助 Mixup 来充分利用增强样本；最后通过实验验证了 RoPDA 的优越性和通用性。

第四章介绍了本文提出的基于思维链与示例选择的上下文学习算法 CoTIS-NER，首先根据上下文学习在命名实体识别任务上存在的挑战引出研究动机；然后详细阐述了 CoTIS-NER 的设计细节，在此基础上将数据增强算法 RoPDA 纳入 CoTIS-NER 的整体框架中；最后通过实验验证了所提出算法的有效性。

第五章主要介绍了命名实体识别系统的搭建过程，首先介绍系统的相关背景，然后详细介绍系统的设计与实现，并对系统的功能进行效果展示。

第六章对全文的研究工作进行了总结与回顾，并针对本文方法的潜在局限进行了讨论，同时展望了未来的可改进方向。

此外，附录A中列出了上下文学习中使用的部分参考提示。

第二章　相关工作

本章首先介绍了命名实体识别任务的基础知识以及基本的序列标注模型，随后阐述了基于 Transformer[33]的预训练语言模型，回顾了几种典型的预训练语言模型并对新兴的大型语言模型进行介绍，最后详细讨论了自然语言处理中的两种学习范式：提示学习和上下文学习。

2.1　命名实体识别

2.1.1　任务介绍

命名实体识别任务的目标是从非结构文本中自动识别出具有特定意义的命名实体，如人名、地名、日期和数量等，不仅需要定位出实体的位置，还需要将实体分类为预定义的类别。命名实体识别广泛应用于自然语言处理系统中，是问答、信息检索和知识图谱提取等下游任务的关键组成部分。命名实体识别有多种建模方式，最常见的方式包括基于序列标注的方法[34]、基于跨度（span）的方法[35-36]以及生成式方法[37]。

序列标注方法是一种最经典的建模方式。在序列标注任务中，给定一段输入的文本序列 $X = \{x_1, x_2, \cdots, x_n\}$，其中 x_i 表示文本序列中的第 i 个单词，n 表示文本序列的长度，需要输出一个标签序列 $Y = \{y_1, y_2, \cdots, y_n\}$，其中 $y_i \in D$ 代表 x_i 对应的标签，D 为标签集合。序列标注是一类非常广泛的任务，可以解决一系列对字符进行分类的问题，如分词、词性标注、命名实体识别和关系抽取等。通常来说，在序列标注任务中首先需要定义标注策略，即使用什么样的格式来对序列进行标注。命名实体识别中常用的标注方式为 BIO 和 BIOES。在 BIO 格式中，B（Begin）代表一个实体的开始，I（Inside）代表一个实体的中间部分，O（Other）代表非实体，即不属于任何预定义实体类型。BIOES 格式是在 BIO 格式

的基础上，新增了标签 E（End）和 S（Single），其中 E 表示一个实体的结束位置，S 表示仅含一个单词的命名实体。图2-1中给出了 BIO 和 BIOES 格式所对应的标注示例。以 BIO 格式为例，对于某个具体的实体类型 X，采用 B-X 和 I-X 来分别表示类型为 X 的实体的开始部分和中间部分，若某个命名实体识别任务的预定义实体类型有 N 种，则标签集合 D 的大小为 $2 \times N + 1$。

图 2-1 BIO 与 BIOES 标注示例

序列标注模型的优势在于简单易实现，但是只能为每个单词分配一个标签，因此无法解决嵌套的实体结构。为了解决这一问题，Sohrab 等人[38]提出了基于跨度的方法，通过遍历所有可能的跨度，并预测每个跨度的类型，来确定文本中实体的位置和类型。在此基础上，Shen 等人[39]提出了一种两阶段的识别方法，首先通过过滤器和边界回归器来生成候选跨度，随后将其分类到相应的类别，这不仅充分利用了实体的边界信息，还大大降低了遍历跨度的计算成本。Li 等人[40]提出将 NER 任务重新表述为机器阅读理解（MRC）任务，首先将对每种实体类型的提取转化为自然语言查询，随后按照 MRC 的方式来将实体作为答案跨度进行提取。

近年来，通过序列生成的方式来提取实体变得越来越流行。Yan 等人[37]将 NER 任务表述成实体跨度序列生成任务，借助于 seq2seq 模型 BART[41]模型和索引指针机制[42]来直接生成实体序列。Cui 等人[43]提出了一种基于模板的生成式方法，首先为每种实体类型创建提示模板，随后通过计算候选跨度在每个模板下的生成概率来确定实体类型。随着预训练语言模型生成能力的不断提升和大模型的提出，现如今的模型能够以无需微调的方式直接提取文本中的实体。例如，向 ChatGPT 中输入 NER 任务的具体任务描述和查询文本，模型就可以直接输出查询文本中的所有实体以及类型[23,44]。然而，目前大模型在 NER 任务上的表现仍然低于有监督学习，因此一些研究关注于设计良好的提示来提升大模型在该任务上的性能[22-24]。

2.1.2 序列标注基本模型

随着深度学习的发展，目前主流的序列标注模型大多使用神经网络模型，其基本框架可以抽象为图2-2所示的三层结构：特征表示层、特征编码层和标签解码层。特征表示层将输入的离散文本序列转化为连续且稠密的向量；特征编码层从特征表示中抽取出有意义的特征与信息，从而得到文本序列的语义向量；标签解码层则负责根据语义向量解码出文本序列的输出标签。

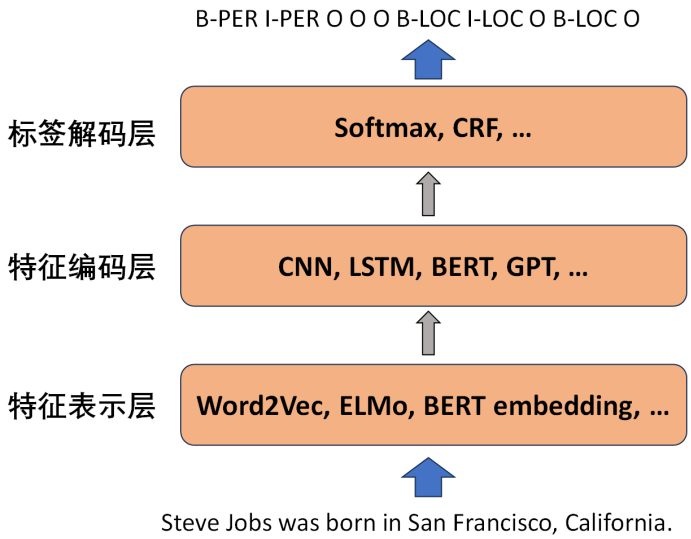


图 2-2 序列标注模型架构

特征表示层最经典的是 Word2Vec[45]和 GloVe[46]词向量，但这类词向量的缺点在于，每个单词只有一个固定表示，无法解决一词多义的问题。为解决这一问题，Peters 等人[47]提出了基于双向 LSTM[48]的 ELMo[47]，可以根据上下文语境来动态调整每个单词的表示。GPT[49]和 BERT[13]等预训练语言模型不仅包含单词的表示（Token Embeddings），还引入了段表示（Segment Embeddings）和位置表示（Position Embeddings），共同为模型提供了丰富的语言表示能力。

特征编码层接收来自于特征表示层的向量表示，并使用 CNN[50]、LSTM 和 Transformer 等结构来抽取文本中的深层语义信息，最终形成融合了上下文信息的语义向量。随着注意力机制的兴起，通过海量语料预训练而获得强大语义理解能力的 BERT 模型脱颖而出，成为 NLP 中最重要的特征编码器。实验证明，Transformer 结构在语义特征提取上的能力显著超越了 CNN 和 LSTM。

标签解码层使用 MLP 结合 Softmax 和 CRF 等结构进行标签解码，将语义向量映射为分类标签。Softmax 是分类问题中最常用的结构，但是在序列标注

任务中，不同位置上单词的标签之间是有联系的，例如 B-Person 后面不能跟着 I-Country，Softmax 却并没有考虑这一关联。CRF 通过学习一个参数矩阵来考虑标签之间的约束转移关系，相比于 Softmax 取得了显著的性能提升。

2.2 预训练语言模型

随着注意力机制的兴起，基于 Transformer 架构的预训练语言模型已经成为 NLP 领域的主流趋势，在各种下游任务上取得了绝对的性能优势[13,49]。预训练语言模型从大规模的语料库中学习通用的语言表示，并在所有的 NLP 下游任务中进行共享。预训练语言模型大致可以分为三类：自回归语言模型、自编码语言模型以及编码器-解码器语言模型[51]。

自回归语言模型（AutoRegressive，AR）根据前面已经生成的文本来预测下一个词的生成概率，采用 Transformer 的解码器结构。AR 的典型代表为 GPT 系列模型[16,49,52]，由于预训练过程与文本生成过程完全一致，具有很强的生成能力，但是只能看到单向的上下文信息，因此内容理解能力稍弱。自编码语言模型（AutoEncoder，AE）在预训练时对原始文本中的单词进行随机掩码，并利用双向的上下文信息进行重建，采用 Transformer 的编码器结构。AE 的典型代表为 BERT[13]和 RoBERTa[53]等，由于能够看到双向的上下文信息，因此擅长自然语言理解任务，但是难以处理生成类任务。编码器-解码器语言模型是一种更加灵活的“输入文本-输出文本”模型，其输入是经过某种特定方式损坏后的序列，输出为重建的原始序列。序列损坏的方式包括文档旋转、句子排列以及单词删除/屏蔽等。编码器-解码器语言模型的典型代表为 BART[41]和 T5[54]，鉴于其序列到序列（seq2seq）的生成性质，常用于机器翻译、摘要生成和风格迁移等任务中。本节将对这三类预训练语言模型的典型代表 GPT、BERT 和 T5 模型进行详细介绍，并讨论当前在 NLP 领域占据统治地位的大型语言模型。

2.2.1 GPT 模型

GPT（Generative Pre-trained Transformer）是第一个基于 Transformer 架构的预训练语言模型，其设计初衷是学习一种通用的语言表示，以适应各种下游任务。GPT 基于 Transformer 解码器结构，通过多层堆叠来捕捉更深层次的语言特

征。自注意力机制是 Transformer 架构的核心，与传统的序列模型如 LSTM 相比，它能够更有效地捕获文本中的长距离依赖关系。GPT 采用了一种单向的因果注意力机制，这使得模型在预测文本序列中下一个单词时，仅考虑序列中前面的单词。GPT 模型的架构如图2-3所示。

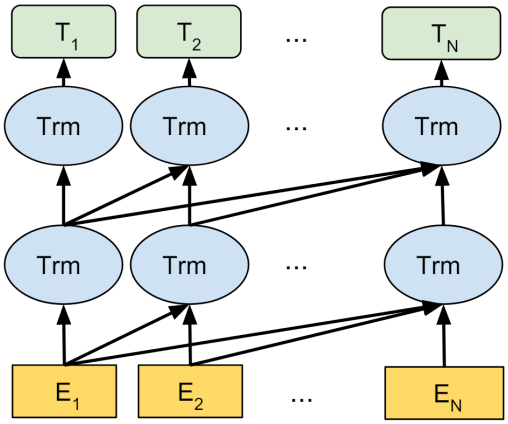


图 2-3 GPT 模型结构[13]

GPT 采用了预训练-微调的两阶段训练范式。在预训练阶段，给定一个文本序列 $U = u_1, \cdots, u_n$，通过最大化以下对数似然来更新模型参数 Θ：

$$L(U) = \sum_i \log P(u_i | u_1, \cdots, u_{i-1}; \Theta),$$

预训练阶段使得模型能够从大规模的文本语料库中学习到通用的语言表示，并具备强大的文本生成能力。在微调阶段，在 GPT 的解码器结构后面添加特定于任务的输出层，并使用下游任务上的标记数据来更新模型参数，从而将 GPT 在预训练阶段获得的丰富语言表示迁移到各种具体的下游任务中，如文本分类、问答系统和机器翻译等。

2.2.2 BERT 模型

BERT 是由 Google AI 提出的一种预训练语言模型，其全称为 Bidirectional Encoder Representations from Transformers。BERT 由多层 Transformer 编码器堆叠而成，与单向语言模型如 GPT 相比，BERT 的显著优势在于其双向上下文理解能力，能够同时考虑输入序列中单词的左侧和右侧上下文信息，因此在自然语言理解任务上表现更为出色。BERT 的整体结构如图2-4所示。

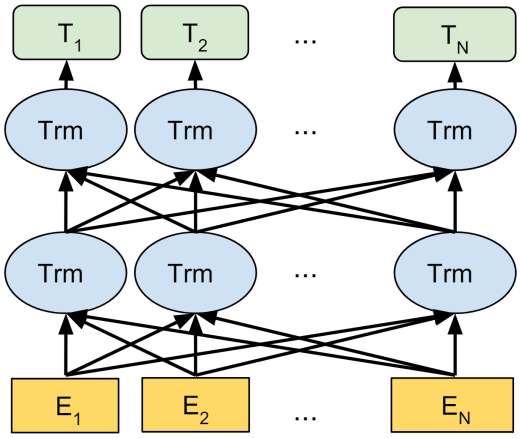


图 2-4 BERT 模型结构[13]

BERT 同样遵循预训练-微调的两阶段训练范式，其预训练过程包括掩码语言模型（Masked Language Model, MLM）和下一句预测（Next Sentence Prediction, NSP）两个任务。MLM 任务将输入文本中 15% 的单词随机替换为掩码标记 [MASK]，并将这些被掩码的单词进行还原，从而帮助模型学习双向的上下文信息。由于微调阶段不包含掩码标记，这导致了预训练与微调阶段之间的不一致性。为了缓解这个问题，BERT 在预训练中采用了一种特殊的策略：被掩码的单词在 80% 的情况下被替换为掩码标记 [MASK]，10% 的情况下替换为随机单词，在另外 10% 的情况下保持不变。为了更好地捕捉句子级的特征，BERT 引入了一个二元分类任务 NSP，用于预测两个句子是否是连续的。对于给定的一个句子 A，在 50% 的情况下选取其下一个句子来作为正例，另外 50% 的情况下从语料库中随机采样一个句子作为负例，将这样的句子对作为 NSP 任务中的训练样本。NSP 任务可以帮助模型充分学习句子级别的结构和语义信息，从而在文本蕴涵、问答等需要理解多个句子之间关系的下游任务中表现得更加出色。BERT 的微调阶段与 GPT 一致，在模型中添加特定于任务的输出层，并使用下游任务上的标记数据来更新模型参数，以快速适应下游任务。

在 BERT 模型中，每个单词的最终嵌入表示是由三种不同的嵌入向量相加得到的，包括单词嵌入（Token Embeddings）、段嵌入（Segment Embeddings）和位置嵌入（Position Embeddings）。单词嵌入负责捕捉每个单词的语义信息，段嵌入用于区分文本中的句子边界，而位置嵌入则提供了单词在文本序列中的位置信息，这三种嵌入表示对于模型理解句子结构都是至关重要的。

2.2.3 T5 模型

2019 年谷歌研究团队推出了 T5 模型，其全称为 Transfer Text-To-Text Transformer。该模型一经发布，就在阅读理解、文本分类和摘要生成等诸多基准测试中都取得了最佳性能，成为最强大的预训练语言模型之一。T5 模型的基本思想是提供了一个 NLP 任务的统一框架，在该框架中将每个 NLP 任务都视为“文本到文本”的问题，即将文本作为输入并产生新的文本作为输出，从而可以将同一模型、目标函数、训练流程和解码过程直接应用于所有 NLP 任务上。

T5 采用了标准的 Transformer 编码器-解码器结构。在处理输入序列时，模型首先将单词序列经过模型嵌入层映射为嵌入表示，随后将其传入到编码器中，编码器中的自注意力机制采用全可见的注意力掩码矩阵，因而在对输入文本编码时可以看到双向的上下文信息。在解码器部分，自注意力层采用了单向的因果注意力，只允许模型关注过去的输出文本。另外，解码器的自注意力层后还有一个标准的注意力机制，该机制通过关注编码器的输出来利用输入文本中的信息。与之前的预训练语言模型相比，T5 模型不仅继承了 BERT 的双向上下文注意力机制，还结合了 GPT 的文本生成能力，具有较强的适用性和灵活性。T5 的整体结构如图2-5所示。

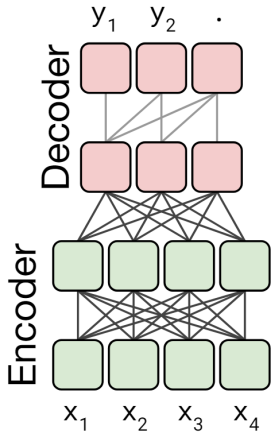


图 2-5 T5 模型结构[54]

研究人员对 Common Crawl 数据进行清洗后获得了较为干净的英文自然文本，并使用该数据来对 T5 模型进行预训练。T5 的预训练任务类似于 BERT 的随机掩码建模，首先对输入文本中的单词进行随机掩码，随后在输出中重建被掩码的标记，其中掩码的比例为 15%。如图2-6所示，与 BERT 不同的是，T5 并不是直接将每个被损坏的标记替换为 [MASK]，而是将每个被损坏的连续跨度整个

替换为一个唯一的掩码标记。另外，解码器无需输出被还原的整个文本，只需要按顺序输出被破坏的跨度即可，这样使得目标序列更短，从而加快训练并减少计算成本。整个训练过程采用极大似然目标来计算损失函数并更新参数。

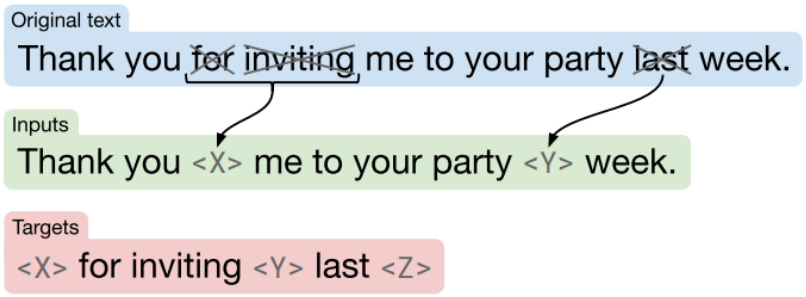


图 2-6 T5 模型预训练建模方式[54]

在处理下游任务时，为了明确指示模型应该执行的具体任务，需要先在原始输入序列中添加一个特定于任务的前缀，然后再将该序列提供给模型。以英文-德文的翻译任务为例，为了让模型翻译句子 “It’s a nice day”，需要将 “translate English to German: It’s a nice day” 作为模型的输入，并以最大似然目标来微调模型使其输出 “Was für ein schöner Tag”。该框架为预训练和微调提供了一致的训练目标，由于所有的 NLP 任务都可以转换为文本到文本的格式，因此 T5 模型可以用来处理所有的 NLP 任务，并且取得了非常优秀的性能。

2.2.4 大型语言模型

2022 年 11 月，OpenAI 发布了基于 GPT 系列模型的会话式模型 ChatGPT，点燃了大模型的研究热潮。大型语言模型通常是指包含数百亿甚至数千亿参数的 Transformer 语言模型[55]，这些模型在海量文本数据上进行了预训练。与其他 PLM 相比，大模型不仅在模型尺寸上大得多，而且还表现出更强的语言理解和生成能力。研究表明，当模型的参数规模到达一定水平时，大模型会表现出涌现能力。涌现能力被定义为 “在小型模型中不存在但是在大型模型中产生的能力”[56-57]，这是大模型与以前的 PLM 最重要的区别。大模型所展现出的涌现能力主要包括（1）上下文学习：大模型在推理时直接从给定的示例中进行学习，无需进行参数调整；（2）指令跟随：对于未见过的新任务，大模型可以在没有使用示例的情况下遵循任务指令；（3）多步推理：大模型可以通过将任务分解为中间

推理步骤来解决复杂问题，比如思维链。大模型的涌现能力使得大模型在各种自然语言处理任务上展现出了令人惊叹的性能，在许多任务上甚至远超人类的表现，比如阅读理解、文本生成和常识推理等。

过去两年中，研究机构发布了许多大模型，其中最广为人知的就是 GPT 家族和 LLaMA 家族。GPT 家族是由 OpenAI 开发的基于 Transformer 解码器的自回归语言模型，包括 GPT-3[16]、ChatGPT 和 GPT-4[58]等。2020 年发布的 GPT-3 有 175B 参数，被广泛认为是第一个大模型，GPT-3 首次展现出了大模型最重要的涌现能力，即上下文学习的能力。ChatGPT 是 2022 年 11 月发布的基于 GPT-3.5 的对话式大模型，它专门针对对话能力进行了优化，在与用户的对话中表现出超强的能力，能够帮助用户解决各种问题。GPT-4 是一个多模态大模型，可以将图像和文本作为输入，并产生文本输出，是目前 GPT 家族中最强大的模型。LLaMA 家族[59-60]是由 Meta 发布的开源语言模型集合，其在 Transformer 解码器架构上进行了一些优化。2023 年，Meta 发布的 LLaMA-2 Chat 模型使用监督微调来对其对话能力进行了优化，是目前最优秀的开源大模型之一。许多研究者们基于 LLaMA 模型进行微调，研发出了 Alpaca[61]、Vicuna[62]等大模型。

2.3 提示学习

随着 BERT、GPT 等预训练语言模型的提出，基于深度学习的自然语言处理从完全监督范式演变为预训练-微调范式。在这种范式中，首先利用自监督任务来对语言模型进行预训练，从大规模的文本数据中学习到通用的语言表示，随后在下游任务的数据上使用特定于任务的目标函数来对预训练模型的参数进行微调。与全监督相比，预训练-微调范式仅需要少量的下游数据来进行微调，减少了对标注数据的需要，并且在下游任务上取得了显著的性能提升。预训练-微调范式通过调整预训练语言模型的参数来使其能够适应特定的下游任务，但由于下游任务与预训练任务之间存在着巨大的鸿沟，无法完全发挥预训练模型本身的能力。此外，尽管微调过程相较于完全监督所需的数据量有所减少，但为了使模型适应特定的下游任务，仍然需要相对较多的数据，从而导致模型在样本稀缺的场景下学习能力差，容易过拟合。

随着 GPT-3 的提出，提示学习（prompt learning）[63]进入了人们的视线，基

于预训练、提示和预测的方法成为了一种新的 NLP 范式。提示学习指通过使用特定的模板来将下游任务的输入输出转化为自然语言描述，从而将所有下游任务统一成预训练任务的形式，实现基于提示（prompt）的统一范式。提示学习一方面缓解了下游任务与预训练任务之间的巨大鸿沟，从而可以帮助模型更好地理解任务，以提高模型的性能；另一方面不需要修改预训练语言模型的结构和参数，因此在少样本或零样本场景下表现优秀。

以二元情感分类为例，下面介绍提示学习的基本流程：

1. 构建模板：首先定义一个由自然语言组成的模板函数 $f_{prompt}(\cdot)$，该模板函数中有两个槽，其中输入槽 [x] 使用输入文本进行填充，输出槽 [MASK] 用于生成答案文本。随后使用输入文本 x 来填充该模板函数，即：$x' = f_{prompt}(x)$。在二分类情感任务中，一个最简单的模板函数为：$f_{prompt}(x) =$ "[x]. It is a [MASK] movie."。例如，对于原始输入文本 x="I love this movie"，经过模板函数映射后，得到 x'="I love this movie. It is a [MASK] movie."；
2. 标签词映射：预先定义一组可能的标签词，并将每个标签词映射到不同的类别，若 1 和 0 分别代表正向和负向情感，则可以定义这样的映射表：$V = \{"good" : 1, "bad" : 0, "excellent" : 1, "horrible" : 0, \cdots\}$；
3. 预测：将 x' 输入到预训练语言模型中，预测映射表 V 中在 [MASK] 位置处概率最大的单词，并根据标签词映射表来将单词转化为分类标签。

提示学习有两种形式，如上面的例子中所示，输出槽位于提示中间的形式被称为完形填空提示，而输出槽位于提示末尾的形式则被称为前缀提示。

提示学习的核心是设计适合特定下游任务的提示模板，从而提升下游任务的性能。创建提示最自然直观的方式是手动编写提示模板。Brown 等人[16]手动创建了前缀提示来处理问答、翻译和常识推理等各种任务。Schick 等人[64-65]在文本分类和条件文本生成任务的少样本学习设置中使用了预定义的手工模板。手动编写提示需要特定于任务的领域知识，且过程较为繁琐耗时，并且模型对构建的提示非常敏感，极易导致性能低下[66]。为了解决这些问题，许多研究致力于自动化模板设计过程。自动生成的模板又可以分为离散提示和连续提示。离散提示即真实的文本字符（tokens），而连续提示则是语言模型嵌入空间中的可学习向量。

部分研究者们通过基于梯度的方式来搜索最佳提示。Wallace 等人[67]设计了一种梯度引导的单词搜索策略，通过迭代更新提示序列中的每个标记，从而查找可以触发 PLM 生成所需目标预测的短序列。在此基础上，Shin 等人[68]提出了 AutoPrompt，首先使用提示模板来处理原始任务输入，随后使用下游任务上的训练样本，利用基于梯度的搜索方法来迭代更新提示中的触发词，在多种下游任务上都展现出了强大的性能。基于梯度的离散搜索方法生成的提示虽然是可读的，但是大多数提示的流畅性和可解释性都比较低。基于释义的方法将现有的种子提示释义为一组新的候选提示，随后选择在目标任务上性能最好的提示。这种释义可以通过多种方式完成，包括反向翻译[66]、借助同义词库来进行短语替换[69]等。一些研究工作将提示的生成视为文本生成任务并利用预训练语言模型来自动生成提示[70-71]，例如，Gao 等人[70]在样本特定位置处插入占位符来表示提示模板，随后使用 T5 模型自动解码出提示词。

离散提示的取值受限于自然语言的范围之内，无法利用反向传播的梯度进行优化，并且具有较大的不稳定性，单个单词的改变就可能导致性能大幅下降[72]。为了解决离散提示的这些局限性，一系列工作专注于优化连续提示[72-77]。Li 等人[73]提出了 prefix-tuning，将一系列特定于任务的连续向量添加到 PLM 每一层输入的前面，同时保持 PLM 的参数冻结，通过反向传播来更新提示向量。P-tuning[72]则仅在输入层添加可训练的连续向量，并且连续向量的插入位置是可选的，不必拘泥于前缀，并利用重参数化来提高收敛速度。为了增强连续提示的可解释性，Passigan 等人[75]通过学习离散提示嵌入的线性组合来获得连续提示。与离散提示相比，连续化的提示向量可以通过高效的反向传播进行优化，具有较高的训练稳定性，展现出了性能优势。与微调相比，连续提示只需要更新很少一部分参数，在少样本环境下表现良好。

2.4 上下文学习

增加语言模型的规模可以在一系列下游任务上带来更好的性能和采样效率，并使得语言模型展现出涌现能力[16,78]。其中上下文学习是大模型最重要的一项涌现能力，其含义为给定自然语言指令和若干个任务演示，与问题拼接起来送入到模型中，模型能够直接输出答案。作为自然语言处理领域的新兴范式，上下

文学习有着十分吸引人的优势：（1）不必对模型进行微调，因此计算成本低并且对计算资源的要求低；（2）不改变模型参数，从而保持了大模型的通用性。大量研究表明，上下文学习存在高度不稳定性，其表现强烈依赖于演示设计。近年来，针对演示设计策略的研究主要分为两类：演示格式的设计和示例的选择。本节将对这两个研究方向进行详细介绍。

2.4.1 演示格式设计

演示格式指模板的结构设计，包括任务指令描述和示例的呈现方式。演示格式的设计旨在寻找一种有效的激励方式来释放大模型的潜力。最简单的演示格式设计是抛弃任务指令，仅将演示示例的问题 x_i 和答案 y_i 与测试问题拼接起来并送入到模型中，但是在一些需要复杂推理的任务上，仅通过若干个演示示例来直接学习从 x_i 到 y_i 的映射并不容易。部分研究者探索了使用大模型来自动化任务指令的生成过程。Honovich 等人[79]发现，在给定几个演示示例的情况下，大模型可以直接生成匹配示例的任务指令。Zhou 等人[80]通过自动提示工程师（Automatic Prompt Engineer）来进行自动指令选择与生成，首先使用 LLM 生成初始提示，随后选择其中准确度最高的提示并生成语义相近的新提示，不断迭代该过程。Pryzant 等人[81]提出基于文本梯度的提示优化方法，在每次迭代中指导 LLM 产生关于旧提示的文本反馈，并用该反馈来更新旧提示。

尽管大模型在自然语言处理领域中展现出了统治性的优势，但是它在数学推理[82]、常识推理[83]和符号推理[25]等复杂推理任务上仍然具有较大的改进空间。研究人员提出通过构建思维链的方式来提升大模型的复杂推理能力，本节随后将对思维链进行详细介绍。

思维链

为了提升大模型在复杂推理任务上的性能，Wei 等人[25]首次提出了思维链（Chain-of-Thought, CoT）的概念。思维链指大模型生成的一系列连贯的中间自然语言推理步骤，其形式为 < 输入 → 推理链 → 输出 >。不同于传统推理直接根据输入得到输出，即 < 输入 → 输出 >，思维链允许模型将一个复杂问题分解为一步一步简单的子问题并依次进行求解，从而得到问题的最终答案，可以显著提

升大模型的性能。

以是否包含上下文演示示例为区分，CoT 可以分为零样本 CoT 和少样本 CoT。Kojima 等人[84]发现，通过在输入中添加一个简单的指令 “Let’s think step by step”，大模型就可以较好地执行零样本思维链推理。少样本 CoT 需要人工给出多个包含中间推理步骤的演示示例。在实际应用中，少样本 CoT 的性能表现要优于零样本 CoT，但是人工编写 CoT 耗时耗力，并且设计与任务相关的推理步骤更是一个困难的任务。为了消除人工设计的成本，Zhang 等人[85]提出了 Auto-CoT 来自动构造演示示例。Auto-CoT 首先将数据集中的问题进行聚类，并从每个簇中选择具有代表性的问题，随后使用启发式零样本 CoT 来为其生成推理链。

原始的 CoT 是使用自然语言描述中间推理步骤的链式结构。很多研究者关注于 CoT 本身的结构问题，将链式的 CoT 转化为表格形式（Program-of-Thought, PoT）[86]、树状（Tree-of-Thought, ToT）[87-88]和图状（Graph-of-Thought, GoT）[89-90]。这种复杂的思维结构能够增强模型解决复杂问题的能力，从而进一步提升模型的推理性能。语言模型的一个问题在于其不可控性，CoT 对中间推理步骤的强调可能在无意中引入幻觉以及累积误差[91]，从而产生不正确的推理路径和答案。因此一类研究致力于通过推理聚合、验证反馈等方法，进一步提升运用 CoT 进行复杂推理的能力。Wang 等人[92]提出了一种自一致性解码策略，以取代思维链中使用的朴素贪婪解码策略。该策略首先从解码器中采样出一组多样化的推理路径集合，随后选择一致性最高的输出结果作为最终答案。在此基础上，Li 等人[93]通过使用多个提示来进一步增加推理路径的多样性，并训练一个验证器来评估推理路径的正确性，随后进行加权投票得到最终答案。Weng 等人[94]通过自我验证来提升推理能力，包含两个步骤：（1）前向推理：对多个候选推理路径进行采样；（2）反向验证：通过掩盖原始条件并预测其结果来计算每个候选答案的验证分数，并根据分数对候选答案进行排序。

对复杂的问题直接求解可能具有挑战性，因此一类方法采用问题分解的方式来提升推理能力。Zhou 等人[95]提出了最少到最多提示（Least-to-Most Prompting）的两阶段策略，其关键思想在于首先将问题分解为一系列更简单的子问题，然后依次解决它们，并通过已经解决的子问题来帮助解决随后的子问题。Dua 等人[96]的方法与之类似，但是他们每次仅分解出下一个需要解决的子问题，再回

答该问题，不断迭代该过程，直至得到最终答案。

目前 CoT 的应用已经从复杂推理任务扩展到了广泛的 NLP 下游任务中。Ma 等人[27]使用 CoT 来完成少样本关系抽取任务，首先诱导大型语言模型使用特定任务和概念级别的知识来生成显式证据，然后将此证据加入思维链提示中以进行关系提取。Wang 等人[26]提出了摘要思维链（SumCoT），旨在指导大模型逐步生成摘要。SumCoT 首先使用手动设计的问题来提示模型从文档中提取核心新闻元素，随后将提取的元素集成起来指导大模型关注更关键的细节，以生成全面的摘要。He 等人[97]研究了将 CoT 用于机器翻译中，有效地提升了翻译性能。Zhang 等人[98]探索了多模态 CoT，将视觉和语言模态纳入一个两阶段框架中，该框架将推理过程和答案生成分离，通过纳入视觉信息来增强模型生成推理路径的能力，并减轻幻觉。

2.4.2 示例选择

研究表明，通过选择不同的示例，上下文学习的性能可以在随机水平到远超人类的水平之间波动。示例选择可以分为两类，一类是在任务级别构建适用于所有测试样例的固定示例集合，另一类则是在示例级别为每个测试样例单独选择出最合适的示例集合。

Zhang 等人[85]发现示例的多样性可以减轻 CoT 中错误的影响，因此提出了基于多样性的示例选择策略：首先将数据集中的问题划分为多个聚类，然后从每个聚类中选择出一个具有代表性的问题。Diao 等人[99]提出了一种基于不确定性度量指标的选择策略，通过选择具有高不确定性的示例来减少模型自身的不确定性，从而提升推理能力。Su 等人[100]提出了一种基于图的选择性标注方法 Vote-K，以选择多样化、有代表性的示例进行标注，大大降低了上下文学习的标注成本。这种为任务选择固定示例集合的方法成本很低，但是对于每个单独的测试样例来说，该示例集合往往不是最优的，从而限制了上下文学习的表现。

部分研究者研究了如何为每个测试样例定制最优的示例集合。Liu 等人[19]观察到与随机选择相比，选择与测试样本语义最相似的 K 个上下文示例能够显著提升性能。在此基础上，Ye 等人[101]通过惩罚与已经选择的示例类似的示例来增加多样性。Ma 等人[102]使用熵来评估每个示例的固有预测偏差，并验证了预测偏差与任务性能之间的关系，选择预测偏差最小的前 K 个示例来构建示例集合。

Wu 等人[29]从信息压缩的视角出发，提出了一个先选择再排序的两阶段架构：首先选择语义相似度最高的前 K 个示例，并随机采样出多个示例组合，以缩小选择空间；随后使用最小描述长度来为示例组合进行排序，以选取最优的组合。总的来说，演示示例的相似性和多样性是示例选择中的两个重要指标，对上下文学习的性能有着重要的影响。

演示示例在提示中的不同展示顺序同样会对 ICL 的性能带来较大的影响。Lu 等人[18]验证了 ICL 的示例顺序敏感性，并提出了一种基于熵的无监督评价指标来评估不同的示例顺序。他们通过从语言模型中采样而构建了一个评估探测集，随后使用全局熵或局部熵指标来评估展示顺序的优劣。

近来关注上下文示例选择的研究工作有很多，然而现有的示例选择方法均是针对句子级任务进行设计的，但 NER 是一个更加关注局部关联的单词级别任务，而非关注句子级语义的句子级任务。针对这一问题，Wang 等人[22]提出了一种基于单词表示来进行最近邻检索的策略，但是他们的方法需要首先微调一个 NER 分类模型，才能得到单词级别的表示。因此，如何为 NER 任务设计更合适的示例选择策略是一个亟待解决的问题。

2.5 本章小结

本章详细介绍了命名实体识别、预训练语言模型、提示学习和上下文学习的相关知识。首先介绍了命名实体识别任务的基本概念以及建模方式，并回顾了该任务中最常用的序列标注模型。随后总结回顾了基于 Transformer 结构的预训练语言模型。然后介绍了自然语言处理的一种学习范式，即提示学习，依次阐述了提示学习的概念、提示模板的形式以及模板搜索的常见方法。最后，本文介绍了大模型中的上下文学习范式，并详细总结了演示格式设计和示例选择这两个重要的研究方向。

第三章　基于数据增强的命名实体识别算法

本章提出了一种基于连续提示的鲁棒数据增强算法（Robust Prompt-based Data Augmentation，RoPDA），旨在为命名实体识别任务生成高质量、高多样性的增强文本，从而扩充标注样本集，提高深度学习模型的泛化性能。

3.1　研究动机

深度学习在自然语言处理领域的成功可以归结于以下三个因素：高容量的模型、计算能力的增强和大规模标注数据的可用性[103]。其中大规模高质量的标注数据可以帮助深度学习模型更好地理解模式和特征，从而提高模型的性能和准确性。尽管 BERT 和 GPT 等预训练语言模型的提出大大降低了模型训练对海量标注数据的需求，但是仍然需要较多的数据来适应下游任务。研究表明，数据量的增加可以持续提升深度学习模型的性能[104]。

命名实体识别是自然语言处理中的一项基本任务，在金融、法律和化学等许多领域都获得了十分广泛的应用，并在一些常见领域上取得了优异的表现。然而不是所有领域都有着大量的标注数据，尤其是法律、化学等高度专业化的领域。由于数据标注需要专家知识并且成本高昂，在这些领域中获取足够数量的标注数据以支持模型训练十分困难。标注数据的数量严重限制了 NER 任务在这些领域中的性能。此外，在大模型时代，虽然大模型的上下文学习能力使得大模型只需要若干个演示示例就可以进行学习，无需大量的标注数据来对大模型进行微调。但是一系列的研究仍然表明，上下文学习对于演示示例的选择十分敏感。在标注样本集容量较小的情况下，可能无法为某些测试示例选择到合适的演示示例，从而导致大模型的性能十分不稳定。Wu 等人[29]的实验结果显示，随着标注

样本数目的增加，上下文学习算法的表现稳步提升。因此可以得出这样的结论：无论是在过去的微调范式下还是在大模型时代，标注样本集的数量和多样性对于下游任务的性能都十分重要。

数据增强通过从已有样本中生成新样本来对样本集进行扩充，是增加样本集数量和多样性的最根本方式。基于规则的方法利用预定义的规则来操作原始文本中的单词，从而生成增强文本[5,7]。这类方法简单直接，但是可能会引入语法错误，并且导致增强后的实体与实体标签不一致。随着预训练语言模型能力的不断提升，研究者们开始探索借助其强大的生成能力来生成增强数据。Zhou 等人[4]使用微调后的掩码语言模型 BERT 来对文本中的实体进行重新生成，从而提升实体多样性。还有一些研究通过使用 seq2seq 类预训练语言模型来生成整个增强文本[3,14]。但这些工作往往存在着增强数据多样性不足、需要借助外界语料库等问题。基于以上考虑，本章克服以往工作的缺陷，借助于预训练语言模型的强大能力，提出了一种鲁棒性强、生成质量好且文本多样性高的数据增强算法。

3.2 算法设计

为了更好地呈现 RoPDA 的算法设计思想，本节首先阐述了 RoPDA 的整体流程，随后详细介绍了数据增强的各个关键模块，涵盖从数据预处理、生成模型的训练，到具体的增强策略，再到增强样本的后处理的整个流程。另外，本节介绍了如何使用 Mixup 来辅助 NER 模型训练，以充分利用增强样本。

3.2.1 整体流程

图3-1展示了 RoPDA 算法的整体工作流程。首先，为了使得预训练语言模型能够更好地理解原始文本中各单词对应的实体类型，并学习单词与类型之间的依赖关系，将原始样本经过一种线性化操作处理为实体与类型相互约束的形式。随后将连续提示向量添加至序列到序列预训练语言模型中，并使用线性化的样本数据对其进行训练。得到训练后的预训练语言模型之后，采用本章中提出的多种基本增强操作来对线性化后的文本进行策略性掩码，并将掩码后的文本送入预训练语言模型中以重新生成增强样本。最后通过自一致性过滤机制来过滤掉包含较多噪声的低质量增强样本。

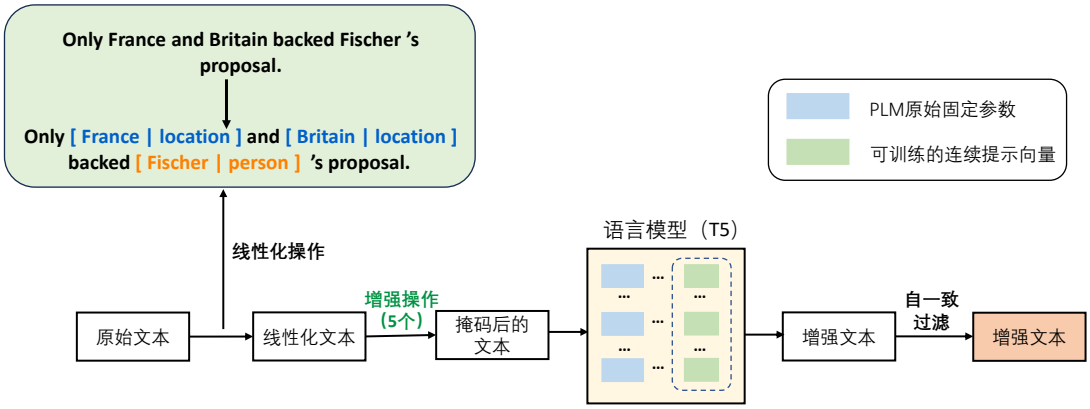


图 3-1 RoPDA 算法整体流程

3.2.2 关键模块

数据线性化

在使用预训练语言模型生成增强数据时，如果仅仅将原始文本提供给模型，模型无法准确判断出文本中实体的位置以及类型，从而难以生成包含实体标签信息的高质量增强文本。为了同时获得增强后的文本以及相应的实体标签，本文提出了一种融合实体标签信息的数据线性化预处理方式，将原始文本和文本中每个单词的实体信息一同提供给 PLM。具体而言，对于给定的文本 $X = [x_1, x_2, \cdots, x_n]$，将文本中类型为 l_k 的实体 $e_{ij} = x_i \cdots x_j$ 转化为 $[x_i \cdots x_j | O(l_k)]$，其中 $O(l_k)$ 表示标签 l_k 的自然语言形式，例如标签 l_k=LOC 的自然语言形式为“location”。图3-1中给出了该线性化预处理操作的示意。在将样本输入到 PLM 之前，需要对每条文本都进行这样的线性化预处理操作。通过使用预处理后的数据来微调 PLM，模型可以更好地理解原始文本中各个单词的实体类型，并学习到单词和实体标签之间的一致性约束。在生成增强数据的过程中，模型能够明确地考虑标签信息，从而约束新实体和实体标签的一致性，提高了生成样本的质量，并为后续的处理任务提供了更加可靠的数据基础。

基于连续提示的模型训练

BERT 等掩码语言模型在生成增强数据时受限于增强前后文本长度必须一致的约束，这极大地限制了生成数据的多样性。因此本章转向序列到序列的语言模型，以期实现更广泛和多样化的数据增强。考虑到 T5 模型卓越的文本生成能

力，本章采用 T5 来作为增强文本生成模型。尽管 T5 模型已经通过预训练积累了丰富的通用知识，但为了使其更好地适应特定下游任务并生成更高质量的增强样本，对其进行进一步的训练仍然是必要的。传统的微调方法通常需要大量的标记数据来更新模型参数，然而在标记数据稀缺的场景下，直接更新所有参数容易导致过拟合，使得模型难以充分学习到下游任务的特定知识和模式。相比之下，提示学习作为一种新兴的学习范式，无需修改预训练语言模型的原始参数，即使在样本量有限的情况下仍然表现优秀，具有较强的通用性。

本章采用了提示学习的方式，通过向模型中引入额外的连续提示向量，来帮助模型更有效地利用标记样本学习下游任务。具体而言，首先在 T5 模型的每个 Transformer 层都添加多个可训练的连续向量。记第 j 层添加的提示向量为 $P_j = [p_{j1}, p_{j2}, \cdots, p_{jk}]$，模型第 j 层的输入为 $H_j = [h_{j1}, h_{j2}, \cdots, h_{jn}]$，在模型的前向传播过程中，第 j 层的自注意力计算可以表示为：

$$\begin{aligned} Q &= [h_{j1}, h_{j2}, \cdots, h_{jn}], \\ K = V &= [h_{j1}, h_{j2}, \cdots, h_{jn}, p_{j1}, \cdots, p_{jk}], \\ head_i &= Self\,Attention(Q, K, V). \end{aligned} \tag{3-1}$$

在训练过程中，仅更新提示向量的参数，而将 T5 模型的所有原始参数冻结。这一策略大幅减少了需要更新的参数数量，不仅有效地避免了过拟合问题，而且显著降低了训练过程中的计算成本和时间消耗，使得整个训练过程更加高效。通过更新提示向量的参数，能够灵活调整模型的行为，并指导其生成特定类型的输出，从而更好地适应下游的数据增强任务。此外，仅更新提示向量还带来了另一个显著优势：每个下游任务只需要保存一组提示向量，并且多个下游任务可以共用同一个 T5 模型，这不仅大大减轻了存储压力，而且提高了模型的可复用性。

在训练阶段，首先采用前面提到的数据线性化操作来对每个训练样本进行线性化预处理。为了保证训练的一致性，本章沿用了 T5 模型预训练阶段的训练方式，即随机跨度掩码（random span mask）。记输入序列为 X，随机采样并丢弃 X 中的连续跨度，每个跨度的平均长度为 3，最终将 X 中约 20% 的单词进行丢弃。本章中采用最大似然损失目标来更新提示向量的参数。

数据增强

在命名实体识别任务中，可以根据文本中的单词是否属于某个实体来将其划分为实体段和上下文段。以往的研究认为，对实体进行增强的收益要远大于对上下文进行增强的收益，因此主要关注于对实体段进行修改或替换以生成增强文本。然而，本文认为同时对上下文进行增强也是必要的。这是因为实体与其周围的上下文紧密相关，通过在替换实体的同时修改周围的上下文，可以使生成的文本更加连贯和合理；此外，多样化的上下文有助于提高样本的多样性，防止模型过度记忆训练数据，并使其学习到更广泛的特征和模式。

给定文本 X，将其切分为 $C_1\,E_1\,\cdots C_n\,E_n\,C_{n+1}$，其中 C_i 代表上下文段，E_i 代表实体段。为了增加数据的多样性，本章提出了五种基本的数据增强操作，这些操作旨在对文本中的实体段和上下文段进行修改，从而改变文本的结构和内容。在进行数据增强之前，需要先使用线性化操作来对输入样本进行预处理，然后使用这五种增强操作来将线性化文本的相应部分进行掩码，随后将掩码后的文本送入训练后的 T5 模型中以重新生成增强样本。图3-2展示了每种增强操作的具体示例，其中图3-2(a) 展示了每种增强操作中的掩码方式，图3-2(b) 展示了每种增强操作生成的增强样本，图中的不同颜色代表了不同的实体类型。下面依次对每种增强操作进行介绍：

- **Op1：增强与实体相关的跨度**。随机选择文本中的一个实体段，并将该实体段与其周围的部分上下文段进行掩码。通过这样的操作，可以修改该实体周围的语义信息，并增强实体多样性。
- **Op2：对实体的类型进行更改**。随机选择文本中的一个实体段，将其实体类型替换为标签集中的其他类型 l_{new}，随后对该实体段及其周围的部分上下文段进行掩码。该操作的目标是改变文本中的实体类型，从而产生不同类别实体出现在该文本中的多样语义。
- **Op3：增加新实体**。随机选择一个实体段 E_i，并在 E_i 的后面添加一个实体类型为 l_{new} 的新实体段及相应的上下文，处理后的文本形式为：$\cdots C_i\,E_i$ $<\text{MASK}>[<\text{MASK}>|O(l_{new})]<\text{MASK}>C_{i+1}\cdots$。该操作的目标是通过引入新的实体来较大地改变文本的语义和结构。
- **Op4：删除实体**。随机选择一个实体段 E_i，并将该实体段及其周围的上下

文段进行掩码，处理后的文本形式为：$\cdots C_{i-1}\ E_{i-1} < \text{MASK} > E_{i+1} \cdots$。该操作将该实体及相关的上下文从文本中删除，同样较大地改变了文本语义及结构，产生与原文本差异较大的新文本。

- **Op5：增强上下文**。随机选择文本中一个长度大于 S 的上下文段，并将该上下文进行部分掩码。该操作注重于对上下文的修改，从而增加了训练集中的上下文多样性。

Op1：	Only [France | location] and [Britain | location] <MASK> [<MASK> | person] <MASK> proposal.
Op2：	Only [France | location] and [Britain | location] <MASK> [<MASK> | organization] <MASK>.
Op3：	Only [France | location] and [Britain | location] backed [Fischer | person] <MASK> [<MASK> | person] <MASK> 's proposal.
Op4：	Only [France | location] and [Britain | location] <MASK> proposal.
Op5：	Only [France | location] and [Britain | location] <MASK> [Fischer | person] 's proposal.

(a) 每种增强操作的掩码方式

Op1：	Only [France | location] and [Britain | location] agree to [Merkel | person] 's proposal.
Op2：	Only [France | location] and [Britain | location] think [European Union | organization] 's decision is unreasonable.
Op3：	Only [France | location] and [Britain | location] backed [Fischer | person] and [John Smith | person] 's farm proposal.
Op4：	Only [France | location] and [Britain | location] agreed to consider their proposal.
Op5：	Only [France | location] and [Britain | location] are not optimistic about [Fischer | person] 's proposal.

(b) 每种增强操作生成的增强数据

图 3-2 增强操作的具体示例

Op1 是标签保留增强操作，将实体重新生成为相同类型的新实体，如北京（Location）→ 纽约（Location），从而增加实体多样性。Op2-Op4 是标签翻转增强操作，它们通过替换实体类型、增加实体以及删除实体来改变原始文本的实体类型序列。从改变实体类型的角度来看，Op2 将某种类型的实体重新生成其他类型的实体；Op3 将一个非实体重新生成为某种类型的实体；Op4 则是将一个特定类型的实体重新生成为非实体。因此概括来说，标签翻转操作可以理解为将文本中类型为 l_{old} 的实体重新生成为 l_{new} 类型的实体。图3-2中展示了标签翻

转操作的效果，经过标签翻转后，增强文本和原始文本的实体类型序列仅在被更改的实体上不同，这种增强文本本质上是实体类型 l_{old} 的对抗性增强示例，可以有效提升 NER 模型区分 l_{old} 和 l_{new} 的能力。本章中新实体类型 l_{new} 是通过从标签集中随机选择而确定的。实验证明，NER 模型可以同时从标签保留和标签翻转操作中受益。Op5 操作通过对上下文进行重新生成来增强上下文的多样性，本文认为这个因素对于增加训练集整体多样性也至关重要。

将这些基本增强操作组合使用可以产生更加多样的增强样本。因此，本文基于这五种操作提出了四种数据增强的策略：标准增强（Standard Augmentation, SA）、实体标签更改（Entity Label Change, ELC）、实体增加（Entity Adding, EA）和实体替换（Entity Replacing, ER）。依据策略中是否包含标签翻转操作，这四种策略可以分为标签保留增强策略（第一种）和标签翻转增强策略（后三种），生成的增强样本也分为标签保留增强样本和标签翻转增强样本两类。每种策略的具体增强操作组成如表3-1所示，以实体替换策略（ER）为例：首先进行 K 次增加新实体和删除实体操作，然后重复执行 $M-K$ 次增强与实体相关的跨度操作，最后执行 N 次增强上下文操作。这里的 K 表示执行标签翻转操作的总次数，M 表示执行标签保留操作的总次数，N 表示执行上下文增强操作的总次数。每种策略都是由标签翻转、标签保留以及上下文增强操作共同组成的，本文认为这三类操作能够产生不同类型的增强样本，结合起来使用可以产生更加多样化的增强数据。其中标签翻转操作可以产生对抗性增强示例，标签保留操作可以增加实体的多样性，而上下文增强操作则可以提升上下文的多样性。

表 3-1 数据增强策略的具体操作构成

增强策略	具体操作
SA	Op1 $\times M$ + Op5 $\times N$
ELC	Op2 $\times K$ + Op1 $\times(M-K)$ + Op5 $\times N$
EA	Op3 $\times K$ + Op1 $\times(M-K)$ + Op5 $\times N$
ER	(Op3 + Op4) $\times K$ + Op1 $\times(M-K)$ + Op5 $\times N$

自一致过滤

通过对上一小节中得到的增强样本进行分析可知，这些样本可能存在着实体类型与其真实类型不一致的问题。例如，某个增强样本中的实体“南京大学”被错误地标注为“location”类型，而实际上应该是“organization”类型。这个问

题在经过标签翻转操作得到的增强样本中尤为严重，这是因为标签翻转操作会改变文本中实体的类型，例如将“location”类型改为“person”类型。这样一来，整个文本的语义就会发生较大的变化，因此生成模型通过修改原始文本来生成通顺流畅的新文本变得更加困难。尽管在标签保留增强操作下，这种不一致性的概率稍低，但仍然存在这个问题。实体类型与真实类型不一致的问题导致增强样本中存在较多噪声，极大地降低了增强样本的质量。为了解决这一问题，本节提出了一种基于自一致性（self-consistency）的过滤策略。该策略首先通过一种双向掩码的机制来构建训练阶段的输入文本，并使用该文本来训练模型，使得模型具备过滤不一致的低质量样本的能力，随后分别通过两个方向上的掩码操作来进行数据增强和数据过滤。

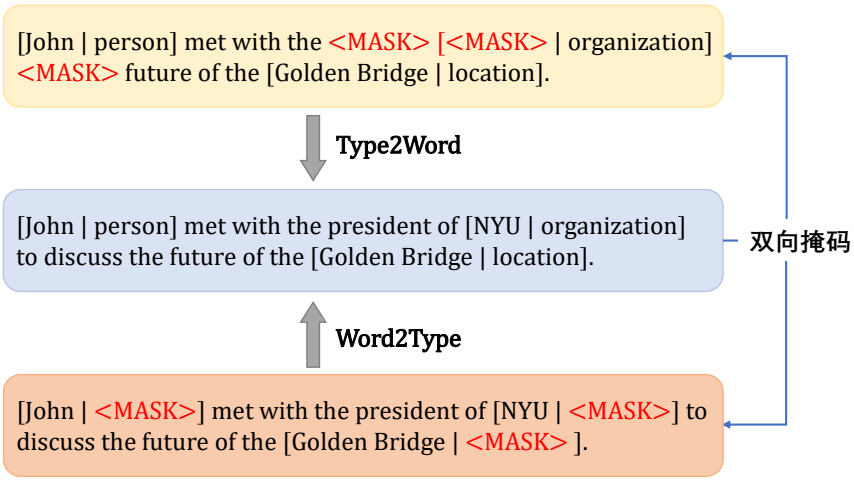


图 3-3 双向掩码示例

如图3-3所示，双向掩码包括 Type2Word 和 Word2Type 两个操作，指分别从两个方向上对线性化后的文本进行掩码操作。Type2Word 指对线性化文本中的单词进行掩码，并根据实体类型来推断出单词。Word2Type 则是指对实体类型进行掩码，并根据单词推断出实体类型。通过使用这两种掩码方式来处理线性化文本，并使用掩码后的文本来对 T5 模型进行训练，可以使得训练后的 T5 模型具备实体/上下文和实体类型之间的双向推理能力，从而能够验证样本是否具备一致性。具体而言，Type2Word 方向上的训练使得模型具备根据前面提出的数据增强操作来生成增强样本的能力。在获得增强样本后，根据 Word2Type 来重新推断增强样本中的每个实体的实体类型，并且仅保留推断出的类型与增强样本

中的原始类型一致的样本。满足这种一致性的样本对于模型来说是自洽的，并且具有较高的置信度。通过采用这种基于自一致性的过滤策略，可以提高增强样本的质量，去除掉噪声较多的样本。

3.2.3 基于 Mixup 的 NER 模型训练

本章使用序列标注的方式来对命名实体识别任务进行建模，并使用 BERT + MLP + Softmax 结构作为基础模型，这是该任务中最常用的模型。使用增强数据的一般方法是将增强样本直接与原始样本混合，然后微调 BERT 模型。前面提到，RoPDA 算法能够生成标签翻转和标签保留两种类型的增强样本，其中标签翻转样本可以看做是一种对抗性增强示例。Lee 等人[105]的研究指出，直接使用对抗样本来训练模型可能会导致对抗性特征过度拟合。而 Mixup 作为一种正则化技术，不仅可以提高模型的泛化性能，还可以提高其对对抗攻击的鲁棒性。因此，本节中提出将标签翻转增强样本与原始样本进行 Mixup，以防止模型过拟合于对抗性增强样本，并提高 NER 模型的泛化能力。

给定样本点 (x, y) 和 (x', y')，其中 x 表示数据点，y 表示标签的 one-hot 表示，Mixup 通过对数据点和标签进行线性插值来创建一个新的样本点，其中混合参数 λ 从 Beta 分布中采样，即 $\lambda \sim Beta(\alpha, \beta)$：

$$\begin{aligned} \hat{x} &= \lambda x + (1-\lambda)x', \\ \hat{y} &= \lambda y + (1-\lambda)y'. \end{aligned} \tag{3-2}$$

由于文本数据是离散的，无法直接在输入空间中进行混合，因此本文中选择在向量表示空间中进行 Mixup。具体而言，在 BERT 模型的第 m 层，将标签翻转增强样本的隐层表示 h_f^m 与原始样本的隐层表示 h_o^m 进行线性插值得到表示 $\hat{h}^m$，然后将 $\hat{h}^m$ 作为输入传递到第 $m+1$ 层中：

$$\hat{h}^m = \lambda h_f^m + (1-\lambda)h_o^m, \tag{3-3}$$

Mixup 的层数 m 从 $\{8, 9, 10\}$ 中随机采样得到，并从一个 Beta 分布中随机采样得到混合超参数 λ。最终的输出标签也是使用相同的超参数进行线性混合。

3.3 实验与分析

为了验证 RoPDA 的有效性与优越性，本节在三个基准数据集上进行了充分的对比实验。实验显示 RoPDA 显著优于目前最先进的数据增强算法，并且在使用无标记数据时优于最先进的半监督学习算法。此外，本节设计了一系列的消融实验来检验 RoPDA 各个模块的有效性。

3.3.1 实验设置

数据集

本节使用了三个来自不同领域、具有不同数量的实体类型的英文数据集，从而对本章提出的RoPDA算法进行全面评估。下面给出了这三个数据集以及4.3.1节中使用的 Wikiann 数据集的介绍。

- **CoNLL03**[106]. CoNLL03 英文数据取自路透社语料库。该语料库由 1996 年 8 月至 1997 年 8 月期间的路透社新闻报道组成，包含 4 种实体类型：PER（人员）、LOC（位置）、ORG（组织）和 MISC（其他）。
- **MIT Restaurant**[107]. MIT Restaurant 由餐厅领域的用户对话组成，包含 8 种实体类型：Price（价格）、Cuisine（烹饪）和 Rating（评分）等。
- **MIT Movie**[107]. MIT Movie 由电影领域的用户对话组成，包含 12 种实体类型：Genre（风格）、Actor（演员）和 Title（标题）等。
- **Wikiann**[108]. Wikiann 英文数据摘录于维基百科文章，包含 3 种实体类型：PER（人物）、LOC（位置）和 ORG（组织）。

表 3-2 数据集统计信息

数据集	领域	实体类型数	训练样本数	测试样本数	实体密度
CoNLL03	新闻	4	14K	3.5K	1.7
MIT Restaurant	餐馆	8	6.9K	1.5K	2.0
MIT Movie	电影	12	8.8K	2.4K	2.2
Wikiann	通用	3	20K	10K	1.4

上述数据集的统计结果见表3-2，其中实体密度指每条训练集样本中实体的平均出现次数。

对比方法

本节将与以下算法进行对比以检验 RoPDA 的性能：

- **Baseline**. Baseline 表示仅使用原始训练样本来训练 NER 模型，而不采用任何数据增强方法。
- **SDANER**[5]. SDANER 提出了多种针对 NER 任务的单词级/实体级替换策略，是 NER 任务中经典的数据增强方法。
- **MELM**[4]. MELM 基于微调后的掩码语言模型 RoBERTa，根据实体标签来预测被掩码的实体，从而生成新实体，提高实体多样性。
- **PromDA**[3]. PromDA 是当前 NER 任务上数据增强性能最好的算法之一，它同样基于加入了连续提示的 seq2seq PLM，并提出了一种双视图数据增强方法，以标签或关键字为条件来生成新文本。
- **MetaST**[109]. MetaST 是一种先进的半监督方法，它通过自训练来利用额外的未标记数据，并通过自适应的样本选择和权重调整来减少由噪声伪标签引入的误差。

实现细节

RoPDA 采用了 T5-Large 模型作为增强数据的生成模型。如3.2.2节所述，训练过程中仅对连续提示的参数进行更新，并冻结 T5 模型的所有原始参数。模型训练采用 Adafactor 优化器，并将学习率设为 1e-3，批量大小设置为 16，模型训练步数设为 3000。生成增强样本时将 K 设置为 1，而 M 和 N 从集合 {1,2,3} 中进行随机采样。

本节将 NER 任务视为序列标注任务，并采用 BERT-BASE 模型作为骨干模型。在 BERT 模型之后添加了一个全连接层和 Softmax 层来完成分类任务。训练 NER 模型时将学习率设置为 5e-5，批量大小设置为 8。Mixup 中的 α 和 β 分别设置为 130 和 5。为了保证比较的公平性，所有数据增强对比方法的 NER 训练配置与 RoPDA 完全相同。所有实验均在不同的随机种子下运行了 3 次，下文中所报告的结果为这 3 次实验的平均值。3.2.2节中提出的 4 种数据增强策略会生成 4 种不同类型的样本，其中之一是标签保留样本，另外三种是标签翻转样本。实验中将这四种策略生成的增强样本混合起来共同用于模型训练，其中标签翻转样

本会按照3.2.3节中提出的策略进行 Mixup。

为了验证 RoPDA 的有效性，本节分别在正常规模训练样本和小规模训练样本的场景下进行了实验。为了模拟不同程度的小样本场景，本节为每个数据集创建了四个小样本环境：shot-5/10/20/50. 在 shot-K 环境下，从原始训练集中为每个实体类型采样 K 个样本来作为训练集，并将剩余的样本添加到未标记数据集中。为了模拟真实的小样本环境，验证集采用同样的方式来采样，并且与训练集的大小一致。以有 4 种实体类型的 CoNLL03 数据集为例，在 shot-50 场景下分别有 200 条训练和验证数据。

算法 1 使用无标记数据来辅助训练 NER 模型

输入: 标记数据集 $\mathcal{T}$; 无标记数据集 $\mathcal{U}$; 置信度阈值 t; 预训练语言模型 LM
输出: 训练后的 NER 模型 M

1: 使用 $\mathcal{T}$ 来对 LM 进行微调
2: 使用 LM 来对 $\mathcal{T}$ 进行数据增强得到 $\mathcal{T}_{a0}$
3: 对 $\mathcal{T}_{a0}$ 进行自一致过滤得到 $\mathcal{T}_{af}$
4: 使用 $\mathcal{T}_{af}$ 和 $\mathcal{T}$ 来共同训练 NER 模型，记为 M_0
5: 使用 M_0 为 $\mathcal{T}_{af}$ 计算置信度，仅保留数值大于 t 的样本，记为 $\hat{\mathcal{T}_{af}}$
6: 使用 $\hat{\mathcal{T}_{af}}$ 和 $\mathcal{T}$ 来共同训练 NER 模型，记为 M_1
7: 使用 M_1 来对 $\mathcal{U}$ 打伪标签，仅保留置信度大于 t 的样本，记为 $\mathcal{T}_u$
8: 对 $\mathcal{T}_u$ 进行数据增强，记为 $\mathcal{T}_{au}$
9: 使用 M_1 为 $\mathcal{T}_{af} \cup \mathcal{T}_{au}$ 计算置信度，仅保留数值大于 t 的样本，记为 $\hat{\mathcal{T}}_A$
10: 使用 $\mathcal{T}, \hat{\mathcal{T}}_A, \mathcal{T}_u$ 共同训练模型得到 M_2
11: $M \leftarrow M_2$
12: 返回 M

无标记数据中包含着丰富的领域知识，如果利用得当可以显著提高模型性能。对比方法中的 MetaST 是当前最先进的半监督学习方法之一，该方法中利用了无标记数据。因此，为了进行公平比较，也为了能够充分验证 RoPDA 的数据增强潜力，本节中提出一种对无标记数据进行数据增强，并利用这部分增强数据来辅助训练 NER 模型的全新方法。算法1中展示了整体训练流程。具体而言，首先在不使用无标记数据的情况下进行数据增强和 NER 模型训练，从而得到模型 M_1（第 1~6 行）。随后使用 M_1 来为无标记数据打上伪标签，并且仅保留具有高置信度伪标签的数据（第 7 行）。之后使用 RoPDA 算法来为这些伪标签数据生成增强数据（第 8 行）。最后，使用所有可用的带标签数据来共同训练 NER 模型（第 10 行）。

评价指标

实验采用 micro-F1 指标来评价算法性能。micro-F1 是衡量多分类模型准确度的最常用指标，同时兼顾了分类模型的精确率（Precision）和召回率（Recall），其定义如下：

$$\begin{aligned} \text{Precision}_{micro} &= \frac{\sum_{i=1}^{n} \text{TP}^i}{\sum_{i=1}^{n} \text{TP}^i + \sum_{i=1}^{n} \text{FP}^i}, \\ \text{Recall}_{micro} &= \frac{\sum_{i=1}^{n} \text{TP}^i}{\sum_{i=1}^{n} \text{TP}^i + \sum_{i=1}^{n} \text{FN}^i}, \\ F1_{micro} &= 2 \cdot \frac{\text{Precision}_{micro} \cdot \text{Recall}_{micro}}{\text{Precision}_{micro} + \text{Recall}_{micro}}, \end{aligned} \tag{3-4}$$

其中 TP^i, FP^i 和 FN^i 分别表示第 i 个类别的 TP（True Positive）, FP（False Positive）以及 FN（False Negative）数目。micro-F1 考虑了各类别的数量，更加注重数据的真实分布。

3.3.2 对比实验

本节对比了 RoPDA 与当前先进的数据增强算法在命名实体识别任务上的性能。通过分别在小规模和正常规模样本场景下进行实验，以及使用无标记数据来辅助训练，验证了 RoPDA 的优越性和通用性。

表3-3的上半部分展示了多个基准数据集上的整体实验对比结果。在没有使用无标记数据的情况下，RoPDA 实现了显著的性能提升。在所有小规模样本的设置下，RoPDA 在全部数据集上都显著优于 SDANER、MELM 和 PromDA，并且在大多数情况下也优于使用额外无标记数据的半监督学习方法 MetaST。与 Baseline 相比，RoPDA 在 CoNLL03 上提高了 2.3 ~8.3%，在 MIT Restaurant 上提高了 0.6 ~4.8%，在 MIT Movie 上提高了 1.4 ~7.5%。与之前的 SOTA 方法 PromDA 相比，RoPDA 在这三个数据集上分别平均提升了 1.6%、1.8% 和 0.7%。

表3-3的下半部分展示了使用无标记数据辅助训练的实验对比结果。这里将按照算法1训练模型称为 RoPDA*，将同样使用无标记数据但是不对无标记数据进行数据增强称为 RoPDA+。与 RoPDA 相比，RoPDA* 在三个基准测试集中分别实现了 1.9%、1.7% 和 1.1% 的平均性能提升。此外，在所有基准数据集上，

RoPDA* 明显优于目前最优的半监督学习方法 MetaST。为了深入探究本章所提出的数据增强算法对无标记数据的影响，本节将 RoPDA* 与 RoPDA+ 进行对比分析。可以看出，RoPDA* 始终优于 RoPDA+，这说明本章提出的数据增强算法不仅对于高质量的数据集有正向的数据增强作用，对于带有较多噪声的伪标签数据集也有着正向的数据增强作用。

表 3-3 小数据规模上的整体实验对比结果

数据集	CoNLL03				MIT Restaurant				MIT Movie			
Shot	5	10	20	50	5	10	20	50	5	10	20	50
Baseline	65.9	73.6	78.5	82.8	50.3	59.2	66.1	70.5	68.0	70.8	76.0	81.2
SDANER	68.7	74.3	79.4	83.4	51.2	59.8	66.2	70.7	72.8	75.6	78.1	81.8
MELM	67.1	74.6	78.1	82.9	50.7	60.1	66.2	70.4	69.2	71.3	76.5	81.5
PromDA	71.1	78.2	81.0	84.2	51.8	60.3	66.7	70.7	75.2	76.4	78.5	82.4
RoPDA	**74.1**	**79.8**	**81.9**	**85.1**	**55.3**	**62.2**	**67.6**	**71.4**	**75.5**	**77.2**	**79.8**	**82.6**
MetaST	70.5	76.4	79.8	83.6	55.2	62.4	68.6	72.5	71.7	77.7	79.0	81.9
RoPDA+	72.1	80.2	84.3	86.2	56.2	62.7	69.1	72.8	72.3	78.0	80.9	83.3
RoPDA*	**75.0**	**81.9**	**85.2**	**86.4**	**56.6**	**64.1**	**69.5**	**73.0**	**75.9**	**78.3**	**81.6**	**83.7**

为了进一步评估 RoPDA 的有效性，本节在正常规模样本的环境下进行了实验。实验结果如表3-4所示，在数据丰富的情况下，RoPDA 在三个基准数据集上同样为模型带来了显著的性能提升。相比于 Baseline，RoPDA 在三个数据集上分别提升了 0.6%、0.7% 和 0.7%。另外可以观察到，一些对比方法在正常数据规模下提升十分微弱甚至没有提升，比如 MELM 在 CoNLL03 和 MIT Restaurant 数据集下的表现反而要比 Baseline 更低。这一现象说明在数据量已经比较丰富的情况下，一些原本表现优异的数据增强方法反而会失效，这进一步体现了 RoPDA 的有效性和数据规模通用性。

表 3-4 正常数据规模上的整体实验对比结果

算法	CoNLL03	MIT Restaurant	MIT Movie
Baseline	90.6	79.9	87.6
SDANER	91.1	80.0	87.8
MELM	90.8	79.7	87.6
PromDA	91.1	80.4	88.1
RoPDA	**91.2**	**80.6**	**88.3**

上述实验可以得出以下结论：(1) RoPDA 在小数据规模和正常数据规模的环境下都显著提升了模型的性能，这体现了 RoPDA 的数据规模通用性；(2) RoPDA 对于标签噪声较多的数据集也有着正向的增强作用，这体现出 RoPDA 的数据质量通用性。以上实验与分析充分验证了 RoPDA 的优越性。

3.3.3 消融实验

本节通过一系列的消融实验来评估各个模块的作用。本节首先分析了连续提示、自一致过滤和 Mixup 对于 NER 模型性能的影响，以及自一致过滤和 Mixup 对每种增强策略的单独影响。其次，本节通过实验评估了每种增强策略的贡献。最后，本节通过对数据增强中的几个重要参数进行调整来找到最佳的参数配置。

模块消融

为了准确量化各个组件的作用，本节分别移除连续提示、自一致过滤以及 Mixup 后进行实验。通过分析表3-5中的实验结果，可以得到以下观察：（1）移除 T5 模型中的连续提示并进行全参数微调导致三个基准数据集上性能显著下降，这说明连续提示相比于标准微调具有更大的优势；（2）去掉自一致过滤后性能平均下降 0.4%，说明该策略能够在一定程度上过滤掉低质量的增强样本；（3）Mixup 的移除同样导致性能有所下降。总体而言，这三个组件都为 RoPDA 的性能优势带来了一定的贡献，移除任一组件都会导致性能下降。

表 3-5 shot-20 环境下各模块移除后的性能对比

算法	CoNLL03	MIT Restaurant	MIT Movie
RoPDA	**81.9**	**67.6**	**79.8**
移除连续提示	80.5	67.1	79.0
移除自一致过滤	81.5	67.1	79.5
移除 Mixup	81.3	67.2	79.4

接下来，本节研究了自一致过滤和 Mixup 对每种数据增强策略的单独影响。本节分别使用每种策略生成的单一增强数据来训练 NER 模型，并在移除自一致过滤和 Mixup 前后进行实验，实验结果如表3-6所示。首先，移除自一致过滤会给每种增强策略都带来一定的性能下降，其中标签翻转策略的性能下降更为明显。特别是实体标签更改策略（ELC）和实体增加策略（EA）的性能下降最多，达到 0.8%。相比之下，标签保留策略的性能下降较小。这是因为标签翻转策略引入了更多的语义和结构改变，从而引入了更多的噪音，因此对于数据过滤的需求更大。这一假设得到了表3-7的数据支持，可以看出，在自一致过滤后，标签保留策略保留了最大比例的增强样本（74.4%），而三种标签翻转策略则过滤掉了更多的低质量样本。

其次，移除 Mixup 对标签翻转策略的影响同样大于对标签保留策略的影响。这是因为对原始数据和对抗性增强示例进行 Mixup 可以防止 NER 模型过度拟合于对抗性特征并提高泛化能力，而对标签保留增强数据进行 Mixup 则不会带来同样的好处。综合以上分析可以得出结论：自一致过滤和 Mixup 均会给 4 种增强策略带来一定的贡献，并且对于标签翻转策略的贡献更大。

表 3-6 shot-20 环境下移除自一致过滤和 Mixup 后每种增强策略的性能变化（以移除前后的 F1 差值度量）

算法	数据集	SA	ELC	ER	EA	所有策略
移除自一致过滤	CoNLL03	0.2	0.8	0.4	0.6	0.4
	MIT Restaurant	0.3	1.0	0.3	1.5	0.5
	MIT Movie	0.2	0.7	0.6	0.4	0.3
	平均值	0.2	0.8	0.4	0.8	0.4
移除 Mixup	CoNLL03	0.2	1.3	0.5	0.9	0.6
	MIT Restaurant	0.1	1.3	0.7	0.8	0.4
	MIT Movie	0.5	0.9	1.2	1.0	0.4
	平均值	0.3	1.2	0.8	0.9	0.5

表 3-7 自一致过滤后每种增强策略下的增强数据保留的数据比例

数据集	SA	ELC	ER	EA
CoNLL03	78.2	63.1	57.5	57.9
MIT Restaurant	72.8	62.3	55.5	48.8
MIT Movie	72.1	61.7	57.7	54.2
平均值	74.4	62.4	56.9	53.6

不同增强策略的组合

为了评估每种增强策略的贡献，本节逐一移除每种策略生成的增强样本后进行实验，实验结果列于表3-8中。当共同使用这四种策略时，模型性能达到最佳水平。另外，移除 ELC 策略时模型性能下降较小，而移除其他三种策略时模型性能有更明显的下降。这表明相较于其他三种增强策略，ELC 的贡献相对有限。这是因为 ELC 策略在生成增强样本时直接修改实体类型，导致文本语义和结构发生了较大的变化，从而引入了较多的噪声。相比之下，尽管 ER 和 EA 也修改了原始文本中的实体类型序列，但是这两种操作是通过增加和删除实体而实现的，相比于直接改变实体类型来说扰动更小。移除 ER 和 EA 这两种策略带来了最多的性能下降，这反映出标签翻转增强样本的重要性。

表 3-8 不同增强策略组合下的性能对比

SA	ELC	EA	ER	CoNLL03	MIT Restaurant	MIT Movie	平均值
✓	✓	✓		81.0	66.8	79.5	75.8
✓	✓		✓	81.3	67.0	79.4	75.9
✓		✓	✓	81.6	**67.9**	79.6	76.4
	✓	✓	✓	81.7	67.4	79.6	76.2
✓	✓	✓	✓	**81.9**	67.6	**79.8**	**76.4**

超参数设置

在生成增强样本时，有几个重要的超参数需要调整。本节通过在 CoNLL03 数据集上进行实验，并通过观察不同超参数取值下的 micro-F1 来确定最佳的超参数组合。超参数 K 表示对每个样本执行的标签翻转操作的数量。实验结果表明，将 K 设为 1 时可以获得最佳性能，而随着 K 的增加，micro-F1 下降了 1.1%。这说明当标签翻转增强操作执行多次时，反而会导致模型性能下降。

超参数 M 和 N 分别表示每条文本进行实体增强和上下文增强的次数。从图3-4中可以观察到，当 $N = 0$ 时，即没有进行上下文增强时，性能下降了 1.2%。这体现出增强上下文的重要性。此外，当 M 和 N 取值过大或过小时，都会对性能产生负面影响。当 M 和 N 从 $\{1, 2, 3\}$ 中随机选择时，可以获得最佳性能。

RoPDA 在执行实体增强时不仅重新生成实体，还对周围上下文进行重新生成，这是因为本文认为该做法可以使生成的实体与其周围上下文更加协调。为了验证这一假设，本节在 CoNLL03 数据集上进行了实验。实验结果显示，仅重新生成实体而不重新生成其上下文时，F1 值下降了 0.8%。这验证了同时重新生成上下文的重要性。

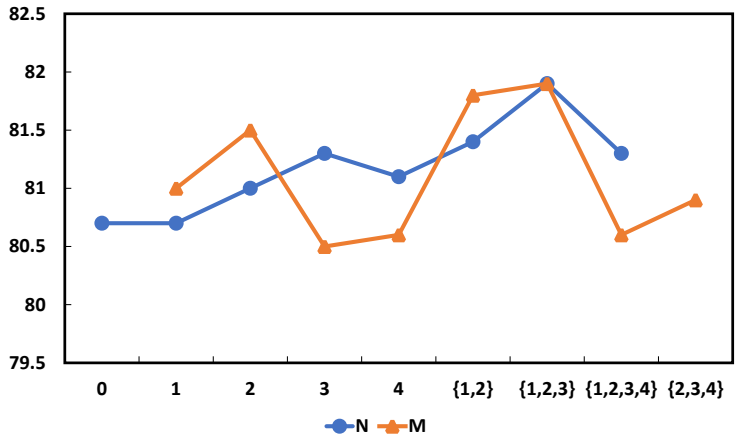


图 3-4 不同 M, N 值在 CoNLL03 数据集上的性能对比

3.3.4 实例分析

为了进一步理解 RoPDA 的特性及优势，本节在表3-9中展示了 RoPDA 生成的一些增强示例。RoPDA 利用预训练语言模型的强大推理能力和蕴含的丰富知识，通过实体与类型的双向约束生成高质量的新实体以增加实体多样性。以 CoNLL03 数据集中的文本为例，“New York”在通常情况下作为地点类型的实体出现，代表一个特定城市，但在本例中，“New York”是一个组织类型的实体，代表一支特定的球队。在 SA 策略中，训练后的语言模型通过上下文推断以及作为约束条件的实体类型，学习到了“New York”的真正含义，并生成了具有类似含义的组织实体“Baltimore Orioles”。在 ELC 和 ER 策略中，原始实体被重新生成为与上下文语义融洽的新类型实体“European”和“Los Angeles”。在 EA 策略中，通过修改原文本的语义和结构，在原始文本中增加了一个新实体“Boston”。此外，RoPDA 不仅提高了实体多样性，还显著增加了上下文多样性。从表3-9的下半部分可以看出，尽管 MIT Restaurant 数据集中的文本整体较短，但除了生成新的高质量实体外，RoPDA 仍为其生成了相对较多样的上下文。

表 3-9 RoPDA 生成的增强数据示例。上半部分数据来自于 CoNLL03，下半部分来自于 MIT Restaurant。加粗部分是数据增强后产生的新实体，加下划线的部分 是数据增强后产生的新上下文。

策略	增强示例
原文本	[Bonds | person] came out of Wednesday ’s game against [New York | organization] in the ninth inning after suffering a mild hamstring strain .
SA	**[Matt Carpenter | person]** was carted out of the game for **[Baltimore Orioles | organization]** in the ninth with a mild hamstring strain.
ELC	**[Federer | person]** pulled out of Wednesday’s **[European | miscellaneous]** final after suffering back strain.
ER	**[Robinson Cano | person]** left the field of Wednesday’s game in **[Los Angeles | location]** with two outs in the ninth after a hard groundout.
EA	**[Marquez | person]** was the **[Boston | organization]** starting pitcher and left the game against [New York | organization]in the ninth after suffering a mild hamstring strain.
原文本	find me a [nice | rating] place to eat that is [not too expensive | price]
SA	find me a [nice | rating] place to have dinner that is **[reasonably priced | price]**
ELC	where is the **[best | rating]** place to get **[chicken wings | dish]**
ER	find me some **[good | rating]** food with **[parking | amenity]** nearby
EA	find me a [nice | rating] place to eat that is [not too expensive | price] and has **[free wifi | amenity]** please

3.4 本章小结

本章提出了一种基于连续提示的数据增强算法 RoPDA。该算法在 T5 模型中引入连续提示向量，并通过更新提示向量的参数来适应下游的数据增强任务。RoPDA 在训练过程中避免了对整个模型进行参数调整，极大地降低了过拟合的风险。RoPDA 的核心优势在于其提出的五种基本增强操作，这些操作分别对实体和上下文进行增强，并生成标签翻转和标签保留的增强样本。为了进一步提升增强样本的质量，RoPDA 采用了一种自一致过滤策略，该策略使用双向掩码来训练 T5 模型，使其具备过滤不一致样本的能力。在 NER 模型的训练过程中，本章将对抗性增强样本与原始样本进行 Mixup，以避免对抗性特征的过度拟合问题。为了全面评估 RoPDA 算法的性能，本章开展了一系列对比实验与消融实验。三个基准数据集上的对比实验结果表明，基于上述思想设计的 RoPDA 适用于各种应用场景，不仅显著优于目前最先进的数据增强方法，并且在使用无标记数据辅助训练时也优于最先进的半监督学习方法。消融实验进一步验证了 RoPDA 中各个模块的贡献和有效性。最后，本章对 RoPDA 生成的增强数据进行展示与分析，证明该算法能够为 NER 任务产生高质量和高多样性的增强样本，充分体现了该算法的优越性。

第四章　基于上下文学习的命名实体识别算法

上一章提出了一种基于连续提示的数据增强算法 RoPDA，能够为 NER 任务生成大量高质量且多样的增强样本，从而解决 NER 任务面临的数据稀缺挑战。本章则关注 NER 任务面临的另一挑战：现有的上下文学习算法在该任务上的表现有待提升。本章为 NER 任务提出了一种基于思维链与示例选择的上下文学习算法（Named Entity Recognition based on Chain-of-Thought and Instance Selection，CoTIS-NER），该算法基于第三章提出的 RoPDA 来对样本集进行扩充，随后通过多步推理来逐步解决问题并通过示例选择来选取最佳演示示例，以充分发挥大模型在 NER 任务上的潜力。

4.1　研究动机

随着大型语言模型的兴起，利用上下文学习来处理各种自然语言处理任务已经成为学术研究的热点之一。上下文学习的性能表现主要取决于演示格式的设计和演示示例的选择，对于命名实体识别任务来说也不例外。

演示格式的设计旨在设计合适的引导方式来充分发挥大模型在下游任务上的潜力。在命名实体识别任务中，精确地识别和抽取实体信息不仅依赖于大模型所积累的丰富知识库，更需发挥其深层次的推理能力。因此，命名实体识别本质上是一个集知识积累与推理分析于一体的任务。然而，现有的上下文学习研究[22-23]往往忽略了该任务对大模型推理能力的需求。图4-1展示了一个使用 ChatGPT 来进行命名实体识别的案例。在此案例中，根据给定的任务描述和上下文示例来以上下文学习的方式引导 ChatGPT 直接输出查询文本中的实体及其类型。该查询文本中的“New York”是一个“organization”类型的实体，特指某

个具体的体育队伍，但是在通常情况下，“New York”作为“location”类型的实体出现，代表一个著名的城市。由于大模型未能深入分析文本语境，并通过必要的推理过程来准确判定实体类型，导致“New York”被错误分类为其更常见的“location”类型。这一简单案例反映了以往的上下文学习算法在 NER 任务上性能较差的一个重要原因——未能充分利用大模型的推理能力来进行准确的分析与预测。

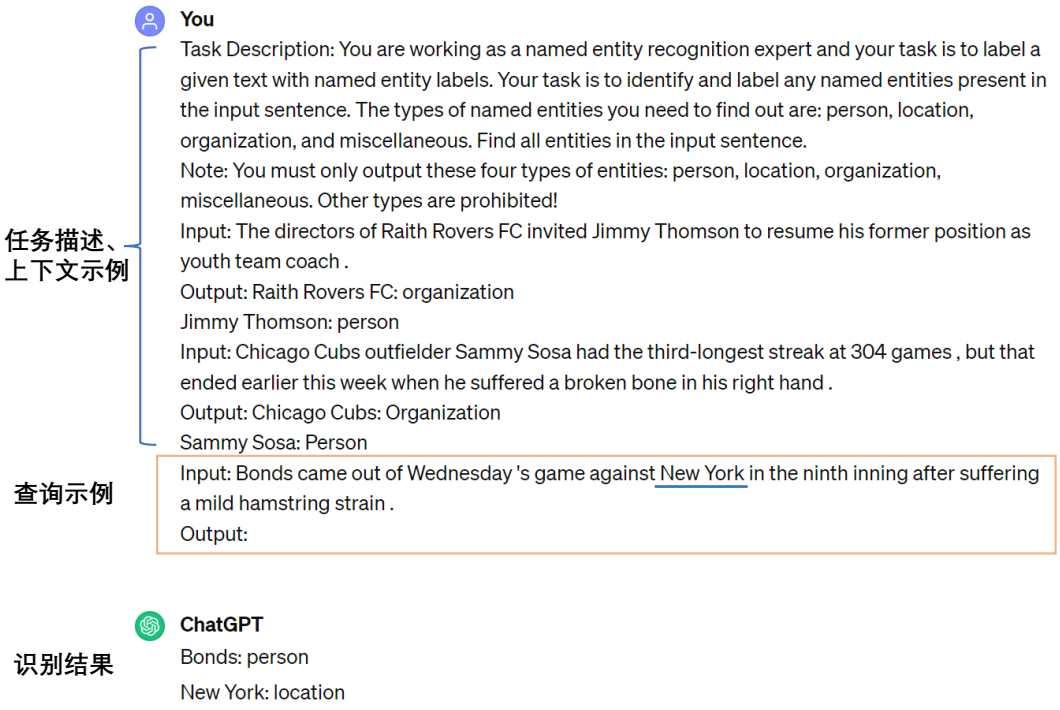


图 4-1 使用 ChatGPT 进行命名实体识别的案例

Wei 等人[25]通过在给出答案之前添加一系列的中间推理步骤来释放大模型的推理能力，这被称为思维链。思维链的引入显著提升了大模型在数学推理和常识推理等需要较强推理能力的任务上的表现。思维链的本质在于逐步解决问题，然而原始的思维链并没有明确地对逐步推理过程进行分离，仍然采用单阶段的生成方式[110]。已有研究指出，将单个复杂问题分解为多个简单子问题并进行逐步求解能够显著降低问题的难度[95]。因此，相比于单阶段的传统思维链来说，通过问题分解来进行多阶段推理可以更好地发挥大模型的潜力。基于以上思考，本章探索通过问题分解和显式多步推理来进行命名实体识别的可能性。

研究表明，选择不同的演示示例会导致上下文学习性能的显著波动。尽管目前已经有许多研究致力于示例选择策略的设计，并在许多下游任务上取得了优

秀的性能表现，但是这些策略大多是为句子级任务设计的，在 NER 这类更加关注局部关联的单词级任务上的表现有待提升。Wang 等人[22]专门为 NER 任务提出了一种单词级别的最近邻示例检索策略，但是该策略需要首先使用标记样本微调一个 NER 分类模型，耗时且繁琐，在实际应用中存在着较大的局限性。因此，为 NER 任务设计合适且高效的示例选择策略，仍然是当前研究中的一个重要挑战。

基于上述背景，本章分别从演示格式的设计和演示示例的选择这两个角度出发，研究如何优化上下文学习在 NER 任务上的性能表现。本章设计了一种以思维链为基础的多步推理方案来完成命名实体识别任务，以充分挖掘大模型的推理能力，并通过同时考虑文本的语义信息和候选实体信息来为测试示例选择更合适的演示示例。

4.2 算法设计

4.2.1 提示设计

通过上下文学习引导大模型完成 NER 任务的一个关键是设计合适的提示。一个好的提示应当准确、具体并且包含足够的背景信息，从而能够让大模型产生符合用户预期的输出。EgoAlpha 团队[111]提出了一个由五个关键部分组成的提示设计框架，其中包括背景介绍、任务指示、参考信息、输出约束和列举例子。本章将参照这一框架来为 NER 任务设计相应的提示。本章在英文数据集上进行实验，但是为了更加清晰地阐述提示设计思路，下面提示构造过程中涉及的提示将使用中文进行描述。

- 背景介绍：通过向大模型提供与指定任务相关的背景信息，能够帮助大模型更好地理解指令含义，有助于生成更加相关和准确的文本。因此本章在提示中向模型提供了“你是一位优秀的命名实体识别专家”的背景介绍。
- 任务指示：任务指示应该清晰、简明和具体。提示中首先给出 NER 任务的具体目标，例如，“你的任务是找到输入文本的所有命名实体及每个实体对应的类型，并且进行输出”，其次给出 NER 任务的实体类型列表，例如，“你需要识别的实体有以下三种类型：任务、地点和组织”。

- 输出约束：输出约束向大模型明确必须做和禁止做的事，有助于生成更加符合意图的输出。首先在提示中添加输出格式的限制，例如，“必须以‘实体：类型’的格式来输出每个命名实体，并且使用逗号进行分隔”。其次添加识别类型的限制，例如，“你只能输出类型属于人物、地点和组织之一的实体，禁止输出其他类型的实体”。
- 列举例子：列举例子可以让大模型更加准确地理解任务意图以及输出格式，从而生成与用户期望一致的文本。本章中为例子的列举设计了合适的示例选择策略。

根据以上思考，本文为 NER 任务设计了合适的提示，但是由于篇幅的限制，本文中给出的提示略去了一些细节，仅保留了最关键的部分。

4.2.2 多步推理

本节为 NER 任务设计了一个基于显式思考的多步推理方案。需要特别强调的是，这里的每一步是指与大模型进行一次交互并完成一次文本生成的过程。该方案包括三个推理步骤，图4-2中给出了每个推理步骤使用的提示以及模型输出的细节。

- 步骤一：大模型推断输入文本中所有可能的候选实体。
- 步骤二：大模型依次对每个候选实体进行分析与推理，输出该候选是否是实体以及属于哪种实体类型的显式思考过程。
- 步骤三：在步骤二的显式思考过程的指导下，大模型推断出每个候选实体的实体类型。

命名实体识别任务的关键在于准确识别出文本中的实体及其对应类型，可以分解为实体跨度预测和实体分类两个子问题。其中跨度预测指预测出文本中所有实体的位置，实体分类则是指预测每个实体的所属类型。在传统的单步推理中，不仅需要关注跨度预测的正确性以及全面性，还需要同时关注实体分类的准确性，这是比较困难的。与单步推理相比，本节提出的多步推理方案通过将 NER 任务分解为多个依次推理的简单问题，从而大大简化了问题的难度。步骤一只需要专注于尽可能地列举出所有正确实体，而无需关注其类型。在步骤二中，大模型利用自身的知识以及语义理解和推理能力来对每个候选实体进行分

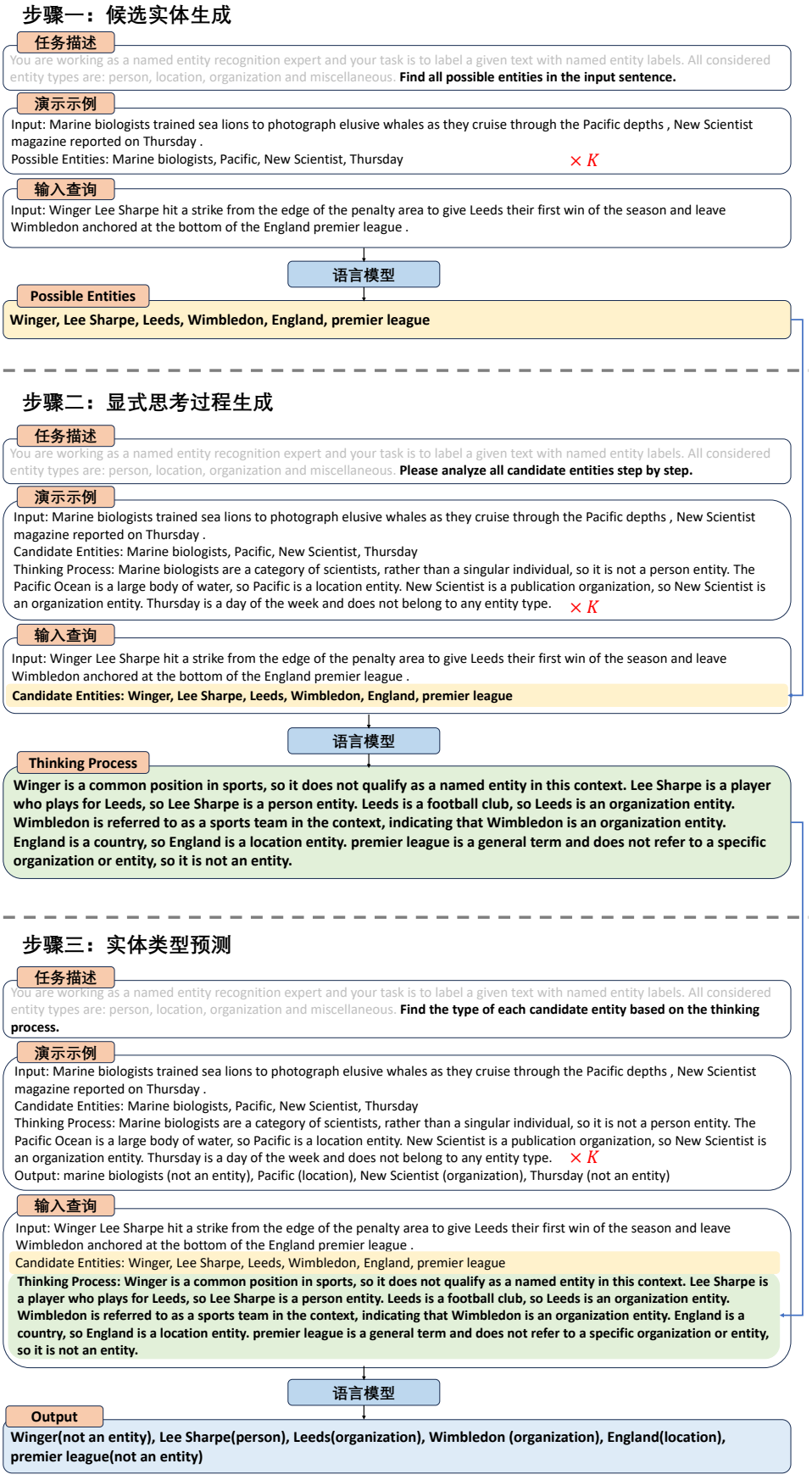


图 4-2 基于显式思考的多步推理过程

析，通过理解每个候选实体的具体含义以及与上下文之间的关系来为候选实体的类型预测提供证据与支持。在步骤三中，根据步骤二输出的显式思考过程来为步骤一中列举出的每个候选实体确定类型并输出最终预测结果。

4.2.3 负样本

多步推理中的步骤一（候选实体生成）是后续推理的基础，某个真实实体是否被准确识别的前提是在步骤一中是否被作为候选实体列出。因此步骤一生成的候选实体列表的全面性至关重要。为了提升实体预测的全面性，本节在多步推理中引入了负样本的概念。负样本即真实类型为“非实体”的候选跨度，一个合格的负样本在判断其是否是实体时应该具有一定难度，从而可以为大模型提供更多的信息。在多步推理中，首先在步骤一中为演示示例引入若干个负样本，随后在步骤二中分析这些候选跨度属于“非实体”的原因。例如，在图4-2中，演示示例中的“Marine biologists”并不是一个真实实体，该跨度的含义为“海洋生物学家们”，指代某个特定的人群，因此将其作为负样本可以帮助大模型更好地理解“person”类型实体的概念，为大模型提供更多的可参考信息。

通过引入负样本，一方面可以使得大模型在进行步骤一推理时，尽可能多地列出查询文本中所有可能的候选跨度，为后续的步骤二推理提供更加全面的实体候选，有助于提升实体预测的召回率；另一方面，大模型在进行步骤二推理时，能够根据上下文示例中的负样本的思考分析过程，来更深入地理解各种实体类型的正确含义与分类边界，从而能够为实体的正确分类提供更多有用的知识。

4.2.4 整体架构

CoTIS-NER 由四个模块构成：支持集构造模块、数据增强模块、示例检索模块和查询推理模块。支持集构造模块旨在为训练样本集中的样本基于其真实标签构造多步推理过程。数据增强模块通过对样本集中的样本进行扩充来增加样本集的多样性。示例检索模块为每个查询示例从支持集中选择合适的演示示例。查询推理模块将任务描述、检索到的演示示例以及输入查询拼接起来形成完整的提示，通过使用4.2.2节中的多步推理方案来引导模型生成最终的预测结果。CoTIS-NER 的整体架构如图4-3所示。下面本节将对每一个模块进行详细介绍。

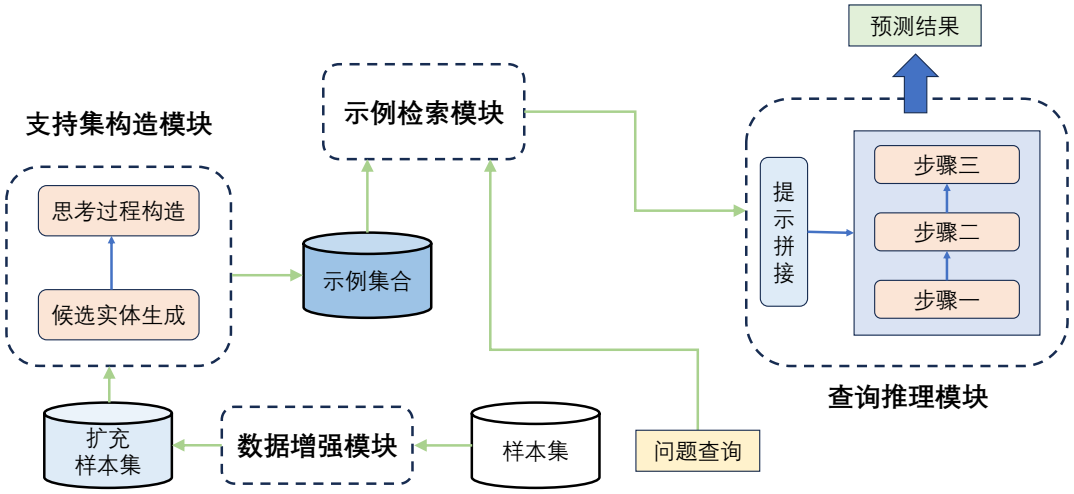


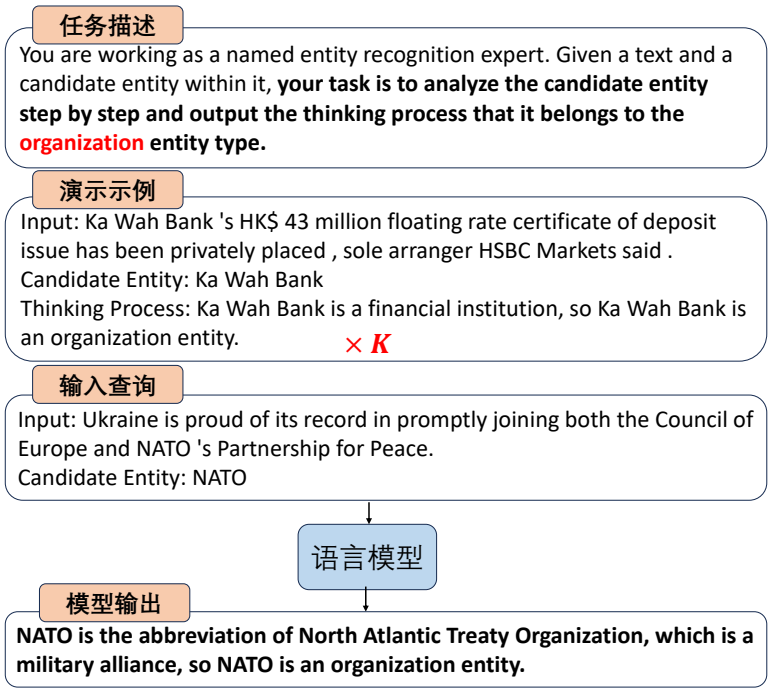
图 4-3 CoTIS-NER 算法整体架构

支持集构造模块

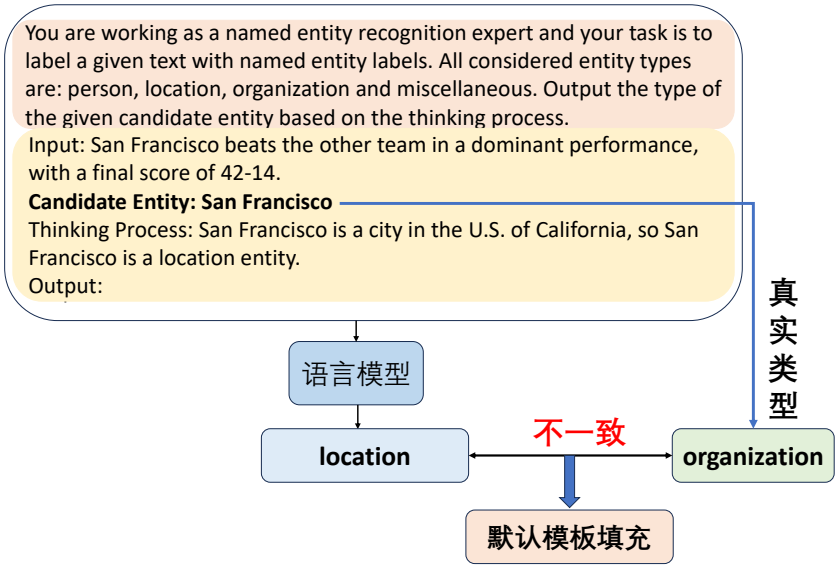
根据4.2.2节中的多步推理方案，构造支持集时需要为训练样本集中的每个样本生成候选实体列表以及显式思考过程。一种常见的支持集构造方式是设计相关模板，通过模板来自动化候选实体以及思考过程的生成。例如，对于实体标签为“person”的实体 **X**，可以通过类似于“In this context, **X** is a specific person, so **X** is a person entity.”这样的模板来自动构造显式思考过程。候选实体列表的构造则可以通过对原文本中的跨度进行随机采样来实现。然而模板构造的方式缺乏灵活性，导致创建的推理过程过于单一和刻板，无法向大模型提供太多有用的信息。跨度随机采样则会使创建的负样本太过简单，同样无法为大模型提供太多指导信息。与固定模板相比，大模型具有强大的生成能力和丰富的知识，因此本节提出了一种使用大模型基于真实标签来为训练样本构造多步推理过程的方案。

1. 自动生成候选实体列表

在构造候选实体列表时，首先从训练样本集 D 中随机选出 m 个样本作为种子示例，并确保每个实体类型在这些示例中至少出现一次。接着，对每个种子示例手动构造候选实体列表，并利用这些种子示例作为演示示例来构造图4-2中步骤一的提示格式，指导大模型为训练集 D 中的每个样本 x 都生成候选实体集合 a，并根据 x 的真实标签 y 来构造真实实体集合 b，将 a 与 b 合并为最终的候选实体列表 c，即 $c = a \cup b$。对于其中的虚假实体集 $d = c - b$（即 c 中不属于 b 的



(a) 显式思考过程生成



(b) 显式思考过程校正

图 4-4 真实标签指导的显式思考过程生成与校正

候选实体），将它们的实体标签设为“not an entity”（即非实体）。

2. 自动生成思考过程

为了提高大模型生成的思考过程的准确性，本节使用真实标签来指导大模型为训练样本中的每个实体生成思考过程。具体而言，对于每个实体类型 e_i，从训练集中选取多个包含 e_i 的样本，手动注释后作为种子示例。随后，利用这些种子示例作为演示示例来构造提示，指导大模型为训练集中的每个 e_i 实体生成该实体属于 e_i 类型的显式思考过程，具体细节如图4-4(a) 所示。

尽管真实标签的指导能够显著提高大模型生成思考过程的准确性，但错误仍然无法完全避免。例如，在图4-4(b) 中，输入文本中“San Francisco”的真实类型为“organization”，然而，由于“San Francisco”在大多数情况下作为“location”类型的实体出现，加之大模型未能准确理解输入文本中的上下文和语义信息，大模型可能会生成错误的思考过程，该思考过程中认为“San Francisco”是一个“location”类型的实体，这种错误的思考会对模型推理产生负面影响。为了解决该问题，本节提出了一种对错误的思考过程进行校正的方案。如图4-4(b) 所示，本节设计相应的提示来指导大模型根据生成的思考过程预测每个实体的类型，如果预测类型与真实类型不一致，说明自动生成的思考过程很可能存在问题，此时将思考过程使用默认模板“In this context, **[Entity]** is a **[Type]** entity.”进行填充，以确保思考过程的准确性。

数据增强模块

在示例检索中，被检索样本集的质量和多样性对于能否为查询示例选择出合适的上下文示例十分重要。因此，CoTIS-NER 在示例检索模块前加入了一个数据增强模块，该模块使用第三章中提出的 RoPDA 算法，通过多种数据增强操作来生成高质量和高多样性的增强文本，从而对原始的训练样本集进行扩充。原始的 RoPDA 需要在 T5 模型中加入连续提示，并进行参数更新。本章使用 LLM 来代替 T5 模型，通过上下文学习的方式来生成增强数据，无需参数调整，具体过程为：首先为数据增强任务编写一套任务描述，并构造若干个上下文示例，其中每个上下文示例由处理后的原文本以及数据增强生成的新文本构成，随后将任务描述与上下文示例以及需要增强的处理后文本拼接起来，输入到 LLM 中以生成增强后的文本。在得到增强文本后，本章仍然通过上下文学习的方式，使用

第三章提出的自一致性过滤策略来对增强文本进行过滤，以提升增强数据的质量。附录A中给出了一份针对数据增强和自一致过滤任务的提示，以供参考。

示例检索模块

研究表明，演示示例的选择对上下文学习的性能具有显著的影响。本节探索了多种示例选择策略，期望为 NER 任务的每条测试示例找到最合适的演示示例。由于基于相似性的示例检索策略已经被证明是十分有效的[19]，因此本章中讨论的几种示例选择策略均基于文本相似性，其计算方式为：给定输入查询 q 和训练集样本 x_i，首先通过文本编码器获得其表示向量 h_q 和 h_i，随后计算其 cosine 相似度 $Sim(h_q, h_i)$。本节探讨的几种示例选择策略的具体说明如下：

算法 2 SDM：同时考虑多样性和相似性来进行示例选择

输入: 输入查询 q，样本集 $D = \{(x_1, y_1), \cdots, (x_m, y_m)\}$，相似度阈值 T，实体类型数 n，最大被选中示例数目 K
输出: 被选中的示例集合

1: 计算 q 和 x_i 的表示相似度 $sim(h_q, h_{x_i})$ 并按照 $sim(h_q, h_{x_i})$ 对样本集 D 进行降序排列
2: $i \leftarrow 1$
3: **while** $i \leqslant m$ 并且示例集合的长度 $<K$ **do**
4: 提取出 (x_i, y_i) 中的实体集合 E_list，类型集合 T_list
5: **if** 已选中示例中的实体类型数 $<n$ **then**
6: **if** T_list 引入了新类型 **then**
7: $mark \leftarrow True$
8: **end if**
9: **else if** E_list 引入了新实体 **then**
10: 计算 x_i 与已选中示例之间的相似度，记最大相似度为 s
11: **if** $s < T$ **then**
12: $mark \leftarrow True$
13: **end if**
14: **end if**
15: **if** $mark = True$ **then**
16: 将 (x_i, y_i) 加入被选中的示例集合中
17: **end if**
18: **end while**
19: 返回被选中的示例集合

- **rand**：从样本集中随机选取 K 个样本作为演示示例。
- **kNN**[19]：选择相似度最高的 K 个样本作为演示示例。

- **Cluster-based**[85]：首先根据向量表示来对训练样本进行 k-means 聚类，随后从每个簇中选取和输入查询相似度最高的样本作为演示示例。
- **SDM(Similarity and Diversity based Methods)**：演示示例的多样性越高意味着可以向大模型提供更丰富的信息。因此除了相似性之外，多样性也是示例选择中一个不可忽视的重要指标。本节提出了一种同时考虑文本相似性和多样性的示例选择策略 SDM。在根据文本相似性选择示例的基础上，SDM 进一步引入了示例之间的语义多样性以及实体多样性，以更有效地满足大模型对丰富信息的需求。SDM 的整体流程如算法2所示。
- **EkNN（Entity-based kNN)**：前述策略均基于原始句子的嵌入表示来进行示例选择，但是句子表示和 NER 任务之间存在着天然的差异：前者是一个句子级别的表示，代表了句子的整体语义信息；而后者则是单词级别的任务，除了整体语义外还十分看重局部的关联。因此仅基于句子的语义表示来选择示例可能会导致次优的效果。考虑到实体信息在 NER 任务中的重要性，本节提出了一种融合实体信息的示例选择策略。具体而言，在多步推理的步骤一中，仍然使用普通 kNN 来进行示例检索，并为输入查询生成候选实体列表。记输入查询 q 的候选实体列表为 l_q，训练样本 x_i 的真实实体列表为 l_{x_i}，随后根据实体列表重构出新的文本 new_q 和 new_{x_i}，并计算 new_q 和 new_{x_i} 之间的相似度，然后选择相似度最高的 K 个训练样本来作为步骤二和步骤三中的演示示例。本节讨论了两种文本重构策略，如图4-5所示，EkNN-1 仅使用实体列表进行重构，EkNN-2 则将原始句子与实体列表拼接起来，同时考虑了句子语义信息和实体信息。

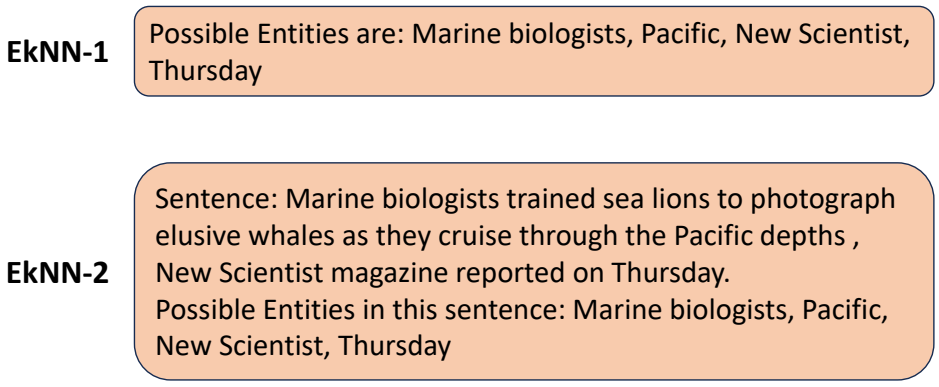


图 4-5 EkNN 中根据实体信息对文本进行重构后得到的新文本

查询推理模块

查询推理模块将任务描述、演示示例以及输入查询拼接起来得到最终的提示，引导大模型生成预测结果。该模块基于4.2.2节中描述的多步推理方案，通过三个步骤依次生成候选实体列表、显式思考过程和实体预测结果。图4-2展示了每个步骤所使用的提示格式。

4.3 实验与分析

为了验证 CoTIS-NER 算法的有效性，本节在三个基准数据集和三个规模的大模型上进行了广泛的对比实验。实验结果显示 CoTIS-NER 的性能表现显著优于以往的上下文学习算法。另外，本节设计了一系列的消融与分析实验来检验 CoTIS-NER 各个模块的贡献。

4.3.1 实验设置

数据集

本节在 CoNLL03、MIT Restaurant 和 Wikiann 这三个英文数据集上进行实验。这三个数据集已在3.3.1节中进行了详细介绍，因此本节不再赘述其内容。

对比方法

本节将与以下算法进行对比以检验 CoTIS-NER 的表现：

- **BERT-tagger**[13]. 使用标记数据来对 BERT 分类模型进行有监督微调，并使用微调后的模型进行测试。
- **Vanilla-ICL**. Vanilla-ICL 设计了一个直接将文本和预测实体拼接起来的提示模板：Input: [text]. Output: [entity1]([type1]), [entity2]([type2])。
- **GPT-NER**[22]. GPT-NER 通过在某种类型的实体前后分别添加特殊标记 @@ 和 ##，来将序列标注任务转化为大模型容易处理的文本生成任务，并通过自我验证策略来解决大模型的过度自信问题。

- **ChatIE**[23]. ChatIE 通过一个两阶段框架来识别命名实体，第一阶段过滤出文本中现有的实体类型，第二阶段通过多轮回合来提取实体，其中每轮仅抽取一种类型的实体。
- **Decomposed-QA**[24]. Decomposed-QA 将 NER 任务按照标签分解为一系列更简单的子问题，在每次问答中只提取一种标签的实体，并通过一种两阶段的多数投票策略来进行一致性改进。

实现细节

本节采用开源的 Vicuna 和 LLaMA 大模型进行实验。为了探索 CoTIS-NER 算法的规模通用性，本节分别在三个不同规模的大模型上进行实验：具有 130 亿参数量的 Vicuna、具有 330 亿参数量的 Vicuna 以及具有 700 亿参数量的 LLaMA 2-Chat，分别简称为 V13、V33 和 L70。其中 LLaMA 2-Chat(70B) 在多个基准测试上的表现与 ChatGPT 相当。本节从 Huggingface[1] 上下载这三个模型的权重，并在具有 8 张 NVIDIA Tesla V100 32GB GPU 的 Ubuntu 系统上进行实验。在示例选择中，本节使用 Sentence-BERT[112] 来进行文本编码，Sentence-BERT 系列包含上百个预训练语言模型，本节选择通用类模型中表现最好的 all-mpnet-base-v2，并从 Huggingface 上下载模型权重。

本节对 CoNLL03 和 Wikiann 数据集进行了随机采样，分别为每个实体类型采样 400 条测试数据，MIT Restaurant 数据集则使用了全部的测试集数据。因此，实验中三个数据集的测试样本数目分别为 1600、1200 和 1518。在上下文学习中，这三个数据集使用的演示示例数目分别为 3、4 和 6。为了更好地对上下文学习和有监督学习进行对比，本节中分别在使用 100 条、200 条和全部训练样本下对 BERT-tagger 模型进行了微调，分别记为 BERT-tagger(#100)、BERT-tagger(#200) 和 BERT-tagger(#All)。本节中的对比实验默认使用 EkNN-2 来选择演示示例。

评价指标

本节使用 micro-F1 指标来衡量算法性能，该指标的详细介绍见3.3.1节，此处不再赘述其内容。

1 https://huggingface.co/

4.3.2 对比实验

本节通过对比 CoTIS-NER 与其他上下文学习算法在命名实体识别任务上的性能，以及 CoTIS-NER 在使用数据增强模块前后的表现，验证了该算法的优越性。

表4-1的下半部分展示了 CoTIS-NER 与其他上下文学习算法的整体实验对比结果。为了更加公平地与其他算法进行比较，这里的 CoTIS-NER 去掉了数据增强的模块。从表4-1中可以得到以下观察：（1）在多数情况下，GPT-NER 的表现比 Vanilla-ICL 更差，这是因为 GPT-NER 在单步推理内不仅需要准确标记出文本中的实体还需要精确还原出原始文本，反而提高了问题的求解难度。（2）GPT-NER、ChatIE 和 Decomposed-QA 这三种算法的性能均没有显著优于 Vanilla-ICL，这说明现有方法仍然有较大的改进空间。（3）与其他方法相比，CoTIS-NER 在所有数据集和大模型上都取得了压倒性的性能优势：相比于 Decomposed-QA 平均性能提升了 13.7%，相比于 Vanilla-ICL 平均性能提升了 14.1%。这证明了 CoTIS-NER 的优越性和模型规模通用性。

为了进一步探究上下文学习算法的表现，本节在表4-1的上半部分中展示了有监督学习算法 BERT-tagger 在不同规模训练样本集下的实验结果。在使用全部训练数据的情况下，BERT-tagger 超越了所有上下文学习算法。尽管 CoTIS-NER 同样比使用全部训练数据的 BERT-tagger 差，但它大幅缩小了上下文学习与有监督学习之间的性能差距。另外可以观察到，当使用 LLaMA 2-Chat(70B) 模型时，CoTIS-NER 的整体表现优于 200 个训练样本下的 BERT-tagger。与 BERT-tagger 相比，CoTIS-NER 仅使用大约 4 个演示示例，并且无需进行任何参数更新，这凸显了 CoTIS-NER 相比传统有监督学习方法的独特优势。

表 4-1 三个数据集和三个大模型上的整体实验对比结果

数据集	CoNLL03			Wikiann			MIT Restaurant		
算法	V13	V33	L70	V13	V33	L70	V13	V33	L70
BERT-tagger(#100)		81.8			67.4			61.5	
BERT-tagger(#200)		83.3			72.6			66.4	
BERT-tagger(#All)		90.7			84.6			79.5	
Vanilla-ICL	49.5	50.0	73.9	55.7	57.7	65.3	49.0	52.9	60.8
GPT-NER	50.1	54.3	61.2	48.3	52.5	59.9	43.8	45.3	51.0
ChatIE	48.7	55.4	64.9	52.3	61.8	50.5	49.6	56.3	61.4
Decomposed-QA	59.2	57.1	65.4	54.3	62.6	54.8	46.0	56.7	61.8
Ours	**69.4**	**71.0**	**83.0**	**69.7**	**71.8**	**76.4**	**64.7**	**67.3**	**68.2**

为了验证数据增强模块的有效性，本节进行了 CoTIS-NER 加入数据增强模块前后的对比实验，并将结果展示在表4-2中。数据增强模块给 CoTIS-NER 带来了 0.4% ~1.0% 的稳定提升，这说明通过对样本集进行数据增强，有效增加了样本集的多样性，有助于为查询文本选择到更加合适的上下文示例。此外，这一现象还说明：尽管大模型的兴起一定程度上降低了对标记数据的需求，但是多样化的标记数据实际上能够对上下文学习中的示例选择阶段起到正向作用，因此在大模型的时代，数据增强技术仍然是有用的。

表 4-2 数据增强模块对性能的影响

数据集	CoNLL03		Wikiann		MIT Restaurant	
算法	V13	V33	V13	V33	V13	V33
CoTIS-NER w/ DA	**70.1**	**71.5**	**70.1**	**72.3**	**65.7**	**68.0**
CoTIS-NER w/o DA	69.4	71.0	69.7	71.8	64.7	67.3

4.3.3 消融实验

本节进行了一系列的实验来评估各个模块的作用。本节首先分析了不同示例选择策略对于实验结果的影响。其次，本节通过将多步推理中的推理步骤进行合并来评估多步推理的必要性。随后，本节分析了使用不同方案来为支持集样本生成推理过程的影响。最后，本节通过组件消融评估了步骤一、步骤二、负样本和上下文示例对 CoTIS-NER 算法的贡献。以下消融实验默认使用 kNN 来进行示例选择。

示例选择策略比较

为了探究示例选择对命名实体识别任务的影响，本节分别使用4.2.4节中提出的多种示例选择策略在 Vicuna(13B) 和 Vicuna(33B) 模型上进行实验，并将实验结果呈现在表4-3中。

可以得到以下观察：（1）每种示例选择策略的性能都显著优于随机示例选择，其性能提升幅度超过 10%，这一发现强调了上下文学习对于演示示例的高度敏感性，并凸显了为下游任务量身定制示例选择策略的重要性。（2）kNN 的性能优于 Cluster-based，由于 kNN 基于语义相似性来选择示例，而 Cluster-based 则是通过聚类的方式来考虑语义多样性，因此这一现象说明仅考虑相似性要比

表 4-3 不同示例选择策略对性能的影响

数据集	CoNLL03		Wikiann		MIT Restaurant		平均值
策略	V13	V33	V13	V33	V13	V33	
rand	51.0	58.8	51.3	61.4	45.0	50.4	53.0
kNN	65.3	70.7	69.1	72.1	63.8	64.7	67.6
Cluster-based	64.3	67.2	66.1	72.5	63.8	65.1	66.5
SDM	66.8	71.0	67.8	**72.8**	**65.0**	65.2	68.1
EkNN-1	61.7	68.3	65.7	70.0	62.4	64.8	65.5
EkNN-2	**69.4**	**71.0**	**69.7**	71.8	64.7	**67.3**	**69.0**

仅考虑多样性更为有效。(3) 本章提出的示例选择策略 SDM 展现出了良好的性能，是除了 EkNN-2 之外最优秀的示例选择策略，该策略综合考虑了原始文本的语义相似性和多样性，证明了在示例选择中平衡这两个因素的有效性。(4) 本章提出的 EkNN-2 几乎在所有基准数据集和大模型上都取得了最佳性能，其平均性能相比排名第二的 SDM 提升了 0.9%。但 EkNN-1 的表现并不理想，仅优于随机示例选择。EkNN-1、EkNN-2 和 kNN 的性能排序为：EkNN-2>kNN>EkNN-1。这三种策略的共同点在于它们都选取相似度最高的前 K 个样本作为示例，但考虑示例相似性的方式有所不同：kNN 只考虑原始文本的语义相似性；EkNN-1 则专注于文本中实体列表的相似性；而 EkNN-2 将这两者结合起来，同时考虑了语义相似性和实体相似性。这一性能排序清晰地表明，在 NER 任务中，结合语义相似性和实体相似性对于提升上下文学习的性能至关重要，另外，语义相似性要比实体相似性更为重要。

综上所述，本章提出的 EkNN-2 在所有示例选择策略中取得了最出色的性能表现。

推理步骤合并

本章介绍的 CoTIS-NER 通过多步推理的方式，逐步引导大模型生成最终的预测结果。在此过程中，CoTIS-NER 与大模型进行三次交互。为了探索多步推理的重要性，本节将其中的几个步骤进行合并，并在 Vicuna(13B) 模型上进行了实验，实验结果呈现在表4-4中。“1,2,3 合并” 表示将多步推理的三个推理步骤合并到一起，引导模型在单次交互中一次性输出候选实体列表、显式思考过程以及最终实体预测这三个步骤的结果。“1,2 合并” 和 “2,3” 合并则分别表示将前两个和后两个推理步骤进行合并。实验结果显示，未经合并的原始 CoTIS-NER 在

所有数据集上都取得了显著的性能优势，与将三个步骤都进行合并相比，性能提升了 6.9%。此外，合并任意步骤都会带来显著的性能下降，而其中将最后两个步骤合并的方案所带来的性能损失相对较小。

表 4-4 推理步骤的合并对性能的影响

方案	CoNLL03	Wikiann	MIT Restaurant	平均值
CoTIS-NER	**65.3**	**69.1**	**63.8**	**66.1**
1,2,3 合并	59.7	57.9	60.0	59.2
1,2 合并	55.9	58.0	58.9	57.6
2,3 合并	60.7	64.3	61.2	62.1

不同支持集示例生成方案的影响

为了高效准确地构建支持集，本章提出了一种利用大模型自动生成推理过程并对其进行校正的新颖方案。本节将与以下四种基准方案进行对比来评估该方案的有效性：（1）方案 1：通过对随机跨度进行采样来构造候选实体列表；（2）方案 2：通过预定义的固定模板来构造显式思考过程；（3）方案 3：仍然利用大模型来构造显式思考过程，但是不使用真实标签进行指导且不进行校正；（4）方案 4：相比于原始方案，去掉了对思考过程进行校正的步骤。本节在 Vicuna(13B) 模型上进行了实验，并将实验结果记录在表4-5中。可以观察到以下结论：（1）相比于原始的 CoTIS-NER，这四种推理生成方案都带来了一定的性能降低。（2）将候选实体列表或思考过程的生成更换为随机采样或固定模板后，算法性能下降了 1.9% 左右，这体现了使用大模型来自动生成推理过程的重要性。（3）在这四种方案中，方案 3 导致了平均 3.7% 的最大性能下降，这说明利用大模型生成推理过程时，如果不使用真实标签来指导生成，反而会给推理过程带来更多的错误，其效果甚至不如使用固定模板的方法。（4）与原始的 CoTIS-NER 相比，移除对思考过程的校正也会带来较小幅度的性能降低，这体现出校正的必要性。以上分析验证了本章提出的支持集示例生成方案的有效性。

组件消融

为了准确量化各个组件的贡献，本节分别将 CoTIS-NER 中的步骤一、步骤二、负样本以及上下文示例移除后进行实验，并在表4-6中给出了三个数据集上

表 4-5 不同推理过程生成方案对性能的影响

方案	CoNLL03	Wikiann	MIT Restaurant	平均值
CoTIS-NER	**65.3**	**69.1**	**63.8**	**66.1**
方案 1	62.7	67.5	62.7	64.3
方案 2	62.4	66.3	63.5	64.1
方案 3	63.1	65.8	58.3	62.4
方案 4	64.9	68.5	63.6	65.7

的实验结果。移除上下文示例带来了最显著的性能下降，与原始 CoTIS-NER 之间的平均性能差异高达 34.7%。特别是在 MIT Restaurant 数据集上，性能下降最为严重，F1 分数不到 20%。这一结果充分反映了上下文学习中演示示例的重要性。另外可以观察到，移除上下文示例后，规模更大的 Vicuna(33B) 模型上的性能反而要比 Vicuna(13B) 更差。此外，移除步骤一和步骤二也分别带来了一定的性能下降，其中步骤一的移除对性能的影响更为显著，这一方面体现出多步推理方案的优势，另一方面表明在多步推理过程中，候选实体的生成步骤要比显式思考的生成更为关键。同样，移除负样本也导致了 CoTIS-NER 性能的显著下降，这说明负样本的引入确实能够为模型提供更多的有用信息，有助于模型进行更准确的推理。综合以上分析可以得出结论：CoTIS-NER 中的这四个组件都十分重要，移除任意一个都会导致一定程度上的性能下降。

表 4-6 各组件移除对性能的影响

数据集	CoNLL03		Wikiann		MIT Restaurant		平均值
算法	V13	V33	V13	V33	V13	V33	
CoTIS-NER	**65.3**	**70.7**	**69.1**	72.1	**63.8**	**64.7**	**67.6**
移除步骤一	59.1	62.7	67.3	69.6	60.2	60.7	63.3
移除步骤二	62.1	69.3	65.7	**72.6**	63.5	64.4	66.3
移除负样本	61.5	67.4	64.5	66.0	62.1	64.4	64.3
移除示例	42.7	42.1	41.7	37.2	18.5	14.9	32.9

4.4 本章小结

本章提出了一种基于思维链与示例选择的上下文学习算法 CoTIS-NER。该算法将 NER 任务分解为三个连续的子问题，并采用逐步推理的方法进行求解。为了提高实体预测的全面性和准确性，CoTIS-NER 在多步推理中特别引入了负样本的推理信息。此外，CoTIS-NER 使用了一种基于真实标签来引导大模型生成支持集样本推理过程的方案，这不仅实现了演示示例构建的自动化，而且有

助于生成更合理、更具信息量的候选实体列表和显式思考过程。考虑到基于原始文本表示的示例选择方法与 NER 任务并不完全契合，CoTIS-NER 采用了一种融合实体信息的示例选择策略 EkNN-2，该策略通过将原始句子与实体列表拼接起来，同时考虑了句子的语义信息和实体信息。本章使用第三章提出的 RoPDA 算法来增强被检索样本集的多样性，这有助于为查询示例选择更为合适的上下文示例。为了全面评估 CoTIS-NER 算法的性能，本章开展了一系列细致的对比与消融实验。三个数据集和三个不同规模大模型上的实验显示，CoTIS-NER 算法在 NER 任务上取得了显著的性能提升。消融实验进一步验证了 CoTIS-NER 中各个模块的有效性。

第五章　命名实体识别系统

为了验证本文所提出算法在实际场景中的有效性，本章搭建了一个命名实体识别系统并将本文提出的两个算法应用其中。下文将对命名实体识别系统的背景、功能、整体架构、实现细节以及识别效果等进行详细介绍。

5.1　相关背景

在信息技术高速发展的时代背景下，命名实体识别在各领域中取得了广泛的应用，可以帮助人们从海量的非结构化文本数据中提取出有价值的信息，进而快速准确地理解文本内容。例如，在金融领域，命名实体识别能够自动识别公司名称、股票代码和交易所名称等重要信息，从而帮助投资者更快速地获取和分析金融资讯；在医药领域，命名实体识别可以帮助医生和研究人员快速识别药物名称、疾病名称和基因名称等关键信息，从而帮助他们更有效地进行医疗诊断和药物研发。在信息爆炸的时代，人们对于文本数据的处理需求日益增长，因此开发一个功能良好的命名实体识别系统的重要性不言而喻，它不仅可以帮助人们更好地理解和利用海量文本数据，还能为各行各业的信息处理工作提供坚实的技术支持。

不同的用户通常具有来自不同领域的命名实体识别需求，而不同领域中的文本和命名实体具有不同的特点和规律。为了满足不同用户的需求，传统的解决方案是为每个应用领域单独开发命名实体识别系统或者模型，这种方法不仅耗时耗力，而且效率低下。相比之下，一个理想的命名实体识别系统应当具备领域通用性，能够准确识别不同领域文本中的各类实体，从而为用户提供更加全面的信息服务。此外，传统的命名实体识别系统通常需要经过长时间的模型训练，用户必须等待模型训练完成之后才能开始使用，这在一定程度上限制了系统的实时性和便捷性。而且，为了满足特定领域的命名实体识别需求，用户通常需要

提供有标记的训练样本来训练模型，但是普通用户往往没有现成的标记样本集，因此如何在有标记数据很少甚至没有的情况下进行命名实体识别也是一个亟待解决的问题。

针对上述挑战，本章设计并实现了一个创新的命名实体识别系统，旨在为所有用户提供更加高效、便捷的命名实体识别服务。不同于以往的系统仅能在某个特定领域上识别实体，该系统具备领域通用与共享的特点，当用户手头没有相关训练数据时，可以通过应用领域的共享实现在多领域上的命名实体识别需求。对于尚未包含在系统中的新领域，用户仍然可以借助于系统中大模型的通用能力来进行实体识别。总的来说，本系统能够全面满足用户在无样本、有未标记样本以及有已标记样本的各种情况下的命名实体识别需求，同时具备快速实时识别的能力，极大地节省了用户的时间和精力。

5.2 系统设计

为了更好地满足人们对非结构化文本数据的处理需求，本章搭建了一个自动化的命名实体识别系统，并将本文第三章提出的数据增强算法 RoPDA 以及第四章提出的上下文学习算法 CoTIS-NER 应用其中，实现了一个多领域通用的命名实体识别系统。

5.2.1 系统需求

本章设计的命名实体识别系统的功能可以概括如下：

- 多领域实体识别：本系统能够在多个预设应用领域上（包括但不限于金融、餐厅、音乐等）高效地执行命名实体识别任务。用户可以通过本系统轻松识别查询文本中的实体，此外，系统提供的在线展示平台将识别结果以直观、易于理解的方式展现给用户。
- 新领域实体识别：面对用户提出的、来自系统尚未覆盖的新领域的命名实体识别需求，本系统能够灵活适应并满足用户的特定需求。
- 新领域创建与共享：本系统不仅支持现有领域的实体识别，还允许用户根据自身需求创建全新的应用领域（如法律），用户构建的新领域不仅为自己所用，还能够与其他用户共享，以提高本系统的领域通用性。

- 实时识别：在处理用户的识别请求时，系统能够迅速响应并实时返回识别结果。用户无需经历漫长的等待时间即可获得所需的信息。

5.2.2 系统架构

根据上节所述的系统需求，本节构建了如图5-1所示的系统架构。该系统整体上可以划分为以下三个模块：（1）数据存储模块：负责数据的存储，数据库中存储着领域相关信息以及样本集数据等。（2）后端处理模块：负责系统功能逻辑的实现，具体包括数据的存取与处理、新领域的创建与领域查找等功能。此外，该模块还集成了高效的命名实体识别算法。（3）前端交互模块：该模块提供了用户与系统交互的界面，对用户行为进行响应，并将后端运行结果展示给用户。数据存储模块和后端处理模块之间通过封装好的数据读写请求进行连接，从而实现数据的存取与处理，前端交互模块和后端处理模块之间通过调用相应的接口函数进行连接。每个模块均承担着特定的功能和责任，共同维护命名实体识别系统的正常运行。

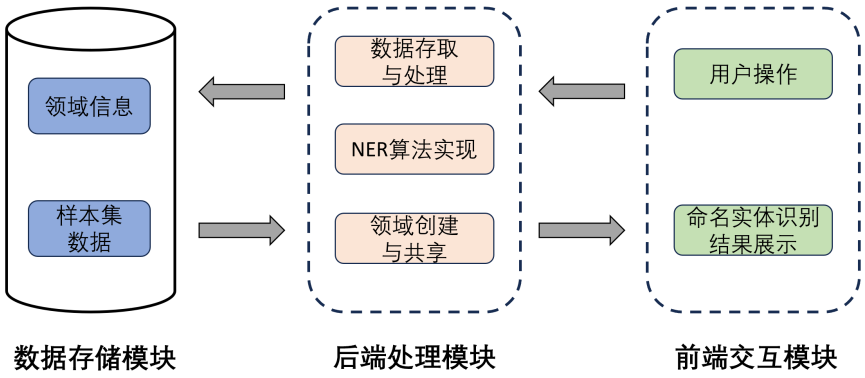


图 5-1 命名实体识别系统架构

5.3 系统实现

根据5.2节所述的系统需求和系统架构，本章实现了一个命名实体识别系统。本节将对该系统的开发环境以及功能实现进行详细阐述。

5.3.1 系统开发环境

在本系统中，后端开发基于 Python 语言中的 Flask 框架，该框架以其轻量级和高效性著称，非常适合快速搭建可靠的后端服务；前端部分采用了标准的 Web 技术栈，包括 HTML、CSS 和 JavaScript；数据存储模块则通过对数据存取操作进行封装而实现。系统中命名实体识别算法的实现基于大模型，对计算资源有较高的要求，因此本节在配备有 NVIDIA Tesla V100 32GB GPU 的 Ubuntu 服务器上部署该系统。在 GPU 资源不足的情况下，本节还提出了一种灵活的替代方案：通过开放的大模型接口（如 ChatGPT）来访问所需的大模型资源。这一策略不仅提高了系统的可扩展性，也保障了系统在多样化环境下的稳定运行。

5.3.2 功能实现

图5-2中给出了本系统的功能选择与实现流程图。用户在首页即可便捷地执行通用的命名实体识别任务。若需针对特定领域进行更精准的实体识别，用户可通过搜索功能，选定相应领域并进行操作。此外，系统还支持用户自定义新领域，并为这些新领域上传相应的样本集。这些功能的结合不仅提升了系统的灵活性，也满足了用户在不同场景下的需求。接下来，本节将对多领域实体识别、新领域实体识别、新领域创建与共享以及实时识别这四个功能的实现进行详细介绍。

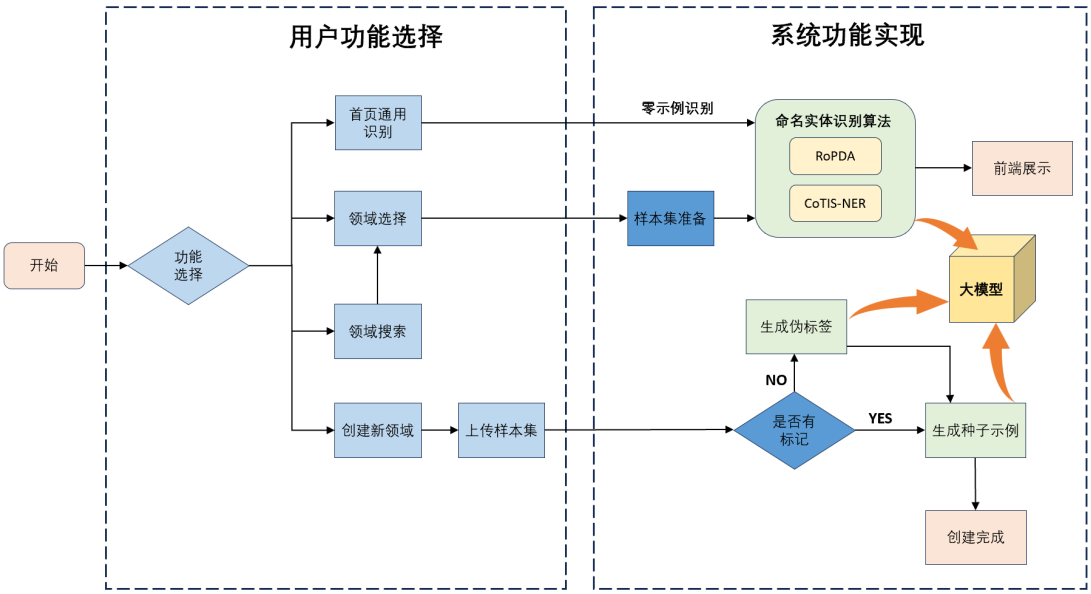


图 5-2 系统整体功能示意

多领域实体识别

大模型通过在海量的文本数据上进行预训练积累了广泛的世界知识，其所具备的通用能力使得大模型可以完成多个领域上的下游任务。本节基于大模型来完成多领域上的命名实体识别功能。使用大模型的好处在于：（1）具备领域通用性；（2）无需进行模型训练，因此系统可以立即为用户返回预测结果，用户不必长时间等待；（3）在没有 GPU 资源的情况下可以通过开放的大模型接口来搭建系统。

为了提升各个领域上命名实体识别的效果，本节预先收集了多个领域的标记数据集用于为上下文学习选择演示示例，如新闻通讯领域（CoNLL03）、餐厅域（MIT Restaurant）和音乐域（CrossNER-music[113]）等，并确定每个领域上的所有实体类型，随后将这些领域设置为系统预设领域，用户可以直接选择在这些领域上进行命名实体识别。随后，本系统基于第三章和第四章中提出的算法来完成多领域命名实体识别的功能。本系统首先使用第三章中提出的数据增强算法 RoPDA 来对标注样本集进行数据扩充，从而增加样本集的多样性，这有助于为查询文本选择到更合适的上下文示例。为了免除数据增强所需的训练过程，不同于第三章中使用 seq2seq 模型 T5 作为生成模型，本章通过设计合适的提示来引导大模型自动生成增强数据。附录A中给出了一份可供参考的提示。在完成数据扩充后，当用户通过前端界面发出实体识别请求时，系统使用第四章中提出的 CoTIS-NER 算法来引导大模型生成实体识别结果，并将结果返回前端，展示给用户。

新领域实体识别

当用户发出系统中尚未涵盖的新领域上的实体识别请求时，由于数据库中缺乏该领域对应的样本集，无法为上下文学习获取相应的演示示例。此时，系统使用 CoTIS-NER 算法在零示例推理的情况下生成识别结果。考虑到相比于有演示示例的情况，零示例推理的效果不够稳定，因此本系统利用大模型的自一致性来提高预测的准确性：首先为查询示例采样出多个推理路径，若某个候选实体在超过一半的推理路径中出现，那么该实体就会被确认为一个有效的预测实体，本系统会统计该实体在所有推理路径中出现次数最多的实体类型，并将其作为

该实体的最终类别。

新领域创建与共享

为了提升本系统的通用性与共享性，系统允许用户通过上传相关样本集来创建一个新领域。用户首先通过前端页面发送创建新领域的请求，并上传一份相关的样本集。为了最大程度地利用数据中的有用信息，系统允许用户上传有标记样本集或无标记样本集。若用户上传了无标记样本集，系统需要先调用零演示示例下的 COTIS-NER 算法来为无标记数据生成伪标签，以供后续使用。4.2.4节中提到，在构造支持集时需要首先为每个样本集选取多个种子示例并手动为种子示例构造推理过程，随后再利用种子示例构建出整个样本集的多步推理过程。然而在本系统中，为每个新领域都手动构造种子示例的推理过程是不现实的，因此本节设计了一种自动为种子示例生成候选实体列表和显式思考过程的方式。本节首先参照4.2.4节中的选取准则来从样本集中选出若干个样本作为种子示例，并使用4.2.4节中为支持集构造推理过程的提示，去掉其中的演示示例部分，来引导大模型为种子示例生成推理过程。

随后，系统在数据库中添加相应的新领域信息，包括领域名称、实体类型、样本集数据和种子示例的信息等。用户创建新领域后，其他用户可以在搜索界面进行搜索，随后系统通过字符匹配的方式来获取一系列相关的领域，并返回给前端进行显示，用户通过点击对应的搜索显示结果来使用该领域，从而实现用户间的领域共享和样本集数据共享。

实时识别

本系统通过利用大模型的上下文学习能力来完成 NER 任务，从而避免了长时间的模型训练。系统提前为每个领域完成数据增强和支持集的构建工作，并使用 Sentence-BERT 模型预先为支持集中的每条示例计算出嵌入表示。当用户发起一次实体识别请求时，系统首先为查询样本计算嵌入表示，随后根据该表示从支持集中检索出合适的演示示例。之后，系统将任务描述、演示示例以及查询文本拼接起来，形成一个完整的提示，引导大模型进行推理并生成预测结果。整个过程完全无需进行模型训练，并且没有十分耗时的计算处理环节，使得系

统能够以极高的效率处理请求并迅速返回识别结果。

5.4 效果展示

本节对搭建的领域通用的命名实体识别系统进行展示。图5-3是本系统的主页面，用户进入该系统后，可以直接在主页面中输入目标语句和需要预测的实体类型，并提交预测。后端处理完成后，前端进行相应的可视化处理，将目标语句中的具体实体情况显示在页面下方。用户还可以通过点击下方的“复制识别结果”和“复制 BIO 标签”按钮来将相应结果复制到剪贴板中。另外，主页面的上方有一个菜单栏，包含了单语句预测和批量预测两个功能。前者是直接输入目标文本并进行预测，后者则是通过上传包含多条语句的文本文件来进行批量预测，最终返回结果文件供用户进行下载。

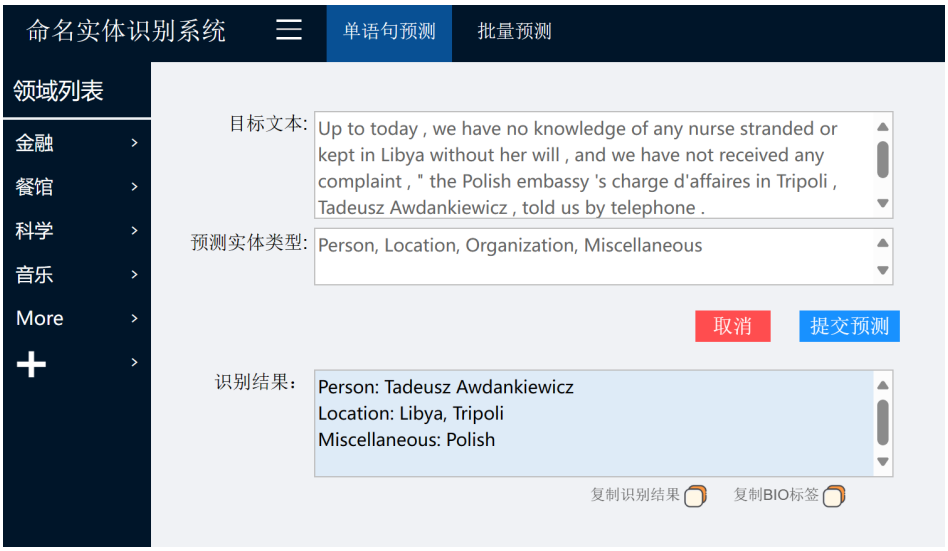


图 5-3 系统主页面

图5-3中的实体识别是在未使用样本集的情况下通过零演示示例推理完成。若用户期望取得更加优秀稳定的识别效果，可以通过页面左侧的导航栏来选择具体的领域，在某个领域下进行实体识别。若左侧的导航栏固定位没有期望的领域，可以通过点击“More”框位来进行搜索。如图5-4所示，用户在搜索框中输入领域名称后，页面下方就会显示出相关领域的信息。用户可以选择按照领域名称的匹配度、使用次数或者创建时间来对搜索结果进行排序。用户可以点击某个感兴趣的搜索结果，随后跳转到对应领域的页面来进行命名实体识别。

图 5-4 领域搜索页面

图5-5中展示了用户在某个具体领域（餐厅）进行命名实体识别的详情。用户首先输入需要处理的目标语句，并确定需要识别的实体类型。每个领域都有着预先定义好的固定实体类型，用户可以选择识别全部类型，也可以仅选定其中的若干个类型进行识别。提交预测请求后，后端将处理结果返回给前端并展示给用户。用户同样可以通过页面上方的菜单栏来选择进行单条语句的预测或者进行批量预测。

图 5-5 在某个特定领域下进行命名实体识别的页面

当用户无法通过搜索框找到所需领域时，用户可以通过点击左侧导航栏的“增加”框位来创建一个新领域。新领域的创建需要用户提供一份样本集，用户

图 5-6 用户创建新领域并上传标记样本集

可以根据自己手中的数据形式来选择上传已标记样本集或者未标记样本集。用户上传一份文本形式的文件后，系统首先对该文件进行检查，并获得其基本信息，例如样本数目、句子的平均长度等。如果用户上传已标记样本集，系统还会统计该样本集中的实体类型数以及各个类型的密度，随后将样本集的详细信息展示给用户，供用户进行确认。如图5-6所示，系统检测出上传的已标记样本集中一共有 4 种实体类型，用户可以确认选择其中的某些实体类型，若用户不选择某种类型，则该类型的实体均会被重新标记为“非实体”。如图5-7所示，如果

图 5-7 用户创建新领域并上传未标记样本集

用户上传未标记样本集，则需要用户手动输入该样本集中的实体类型。随后用户点击确认按钮后，就可以向系统中添加一个新的领域。在添加成功后，所有用户均可以通过搜索界面搜索到该领域并进行使用。

5.5 本章小结

本章详细介绍了命名实体识别系统的设计框架与实现细节。本章首先分析了系统开发的背景，指出了现有命名实体识别系统的不足之处。随后，本章对系统的关键需求进行梳理，并介绍了系统的整体架构以及各个模块的功能。在此基础上，本章详细阐述了系统核心功能的实现细节，并展示了该系统的实际运行效果。本章将第三章的数据增强算法 RoPDA 和第四章的上下文学习算法 CoTIS-NER 应用于该系统中，实现了更加准确和稳定的实体识别功能，充分反映了本文所提出算法的实际应用价值。

第六章　总结与展望

命名实体识别是自然语言处理中的一项重要任务，它广泛应用于信息抽取、对话系统和构建知识图谱等多个下游任务中，发挥着不可或缺的作用。本文针对命名实体识别任务展开研究，聚焦于该任务面临的两大关键挑战：标注数据的稀缺性和上下文学习在该任务上的表现不佳。本文首先从数据的角度出发，研究如何生成高质量且多样化的增强数据，以应对数据稀缺的挑战。随后，本文着眼于大模型的应用，致力于为该任务设计优秀的上下文学习算法，以期充分发挥大模型的潜力。最后，本文设计并实现了一个领域通用的命名实体识别系统，该系统集成了本文提出的两种创新算法，充分展示了本文研究内容的实际应用潜力。本文的具体研究内容与贡献如下：

- 本文从数据的角度出发，提出了一种基于连续提示的数据增强算法 RoPDA。该算法在 T5 模型的每一层中加入了连续提示向量，并通过更新提示向量的参数来适应下游的数据增强任务。RoPDA 在训练过程中避免了对整个模型进行参数调整，有效降低了过拟合的风险，确保模型在样本数量有限的情况下也能充分学习。RoPDA 提出了多种基本增强操作来分别进行实体增强和上下文增强，并生成标签翻转和标签保留的增强样本，从而大大增加了增强数据的多样性。为了进一步提升增强样本的质量，本文提出了一种自一致过滤的策略，该策略使用双向掩码来训练 T5 模型，使其具备过滤不一致样本的能力。经实验验证，RoPDA 能够为命名实体识别任务产生质量高、多样性强的增强样本，从而为后续研究奠定了良好的数据基础。
- 考虑到命名实体识别任务对大模型推理能力的需求，本文创新性地提出了一种基于思维链与示例选择的上下文学习算法 CoTIS-NER。该算法首先将命名实体识别任务分解为三个递进的子问题，随后逐步对每个子问题进行推理。为了向模型提供更多的参考信息并提升实体预测的全面性，CoTIS-NER 在多步推理中引入了负样本推理信息。此外，本文提出了一种利用

真实标签引导大模型为支持集样本生成推理过程的方案，这不仅实现了演示示例构建过程的自动化，而且有助于生成更合理、更具信息量的候选实体列表和显式思考过程。为了帮助测试示例选择到更加合适的演示示例，CoTIS-NER 首先通过 RoPDA 算法来对样本集进行扩充，随后针对命名实体识别任务的特点设计了一种融合实体信息的示例选择策略。经实验验证，CoTIS-NER 在命名实体识别任务上取得了显著的性能提升。

- 本文实现了一个领域通用的命名实体识别系统，并集成了本文提出的两个命名实体识别算法。该系统致力于为用户提供高效、精确的命名实体识别服务，其特点在于具备领域通用性，能够全面满足用户在多个领域下的识别需求，同时确保用户能够即时获得识别结果。

基于本文的研究工作，未来可以在以下几个方向上继续改进：

1. 拓展到更多语言上。命名实体识别任务不仅在英语中发挥着重要作用，在其他语言中也同样应用广泛。目前本文提出的两个算法都只在英文数据集上进行实验，并没有推广到其他语言上。因此未来的研究可以尝试在其他常用语言中应用本文提出的两个算法，从而推动多语言命名实体识别研究的繁荣发展。
2. 大模型的幻觉问题可能导致大模型产生错误的推理步骤并对预测结果过于自信。本文中没有对这一问题进行深入研究与分析。未来研究可以利用大模型的自一致性，对每条查询文本进行多次实体预测，并综合多次的预测来确定最终答案。此外，未来研究还可以考虑利用验证与反馈策略，引导大模型对自己生成的推理步骤给出反馈，并结合反馈来对推理进行改进。
3. 在当前的命名实体识别系统中，不同领域的样本数据无法实现共享与通用。未来可以探索跨领域命名实体识别技术，从而可以有效地利用其他领域的数据资源，进一步提高命名实体识别系统的准确性与灵活性。

参考文献

[1] OpenAI. Introducing ChatGPT[EB/OL]. 2022. https://openai.com/blog/chatgpt.

[2] LI B, HOU Y, CHE W. Data augmentation approaches in natural language processing: A survey[J]. arXiv preprint arXiv:2110.01852, 2021.

[3] WANG Y, XU C, SUN Q, et al. PromDA: Prompt-based data augmentation for low-resource nlu tasks[C]//Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics. 2022: 4242-4255.

[4] ZHOU R, LI X, HE R, et al. MELM: Data augmentation with masked entity language modeling for low-resource NER[C]//Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics. 2022: 2251-2262.

[5] DAI X, ADEL H. An analysis of simple data augmentation for named entity recognition[C]//Proceedings of the 28th International Conference on Computational Linguistics. 2020: 3861-3867.

[6] HU X, JIANG Y, LIU A, et al. Entity-to-text based data augmentation for various named entity recognition tasks[C]//Findings of the Association for Computational Linguistics: ACL 2023. 2023: 9072-9087.

[7] WEI J, ZOU K. EDA: Easy data augmentation techniques for boosting performance on text classification tasks[C]//Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing. 2019: 6382-6388.

[8] ZHANG R, YU Y, ZHANG C. SeqMix: Augmenting active sequence labeling via sequence mixup[C]//Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing. 2020: 8566-8579.

[9] SAWHNEY R, SOUN R, PANDIT S, et al. CIAug: Equipping interpolative augmentation with curriculum learning[C]//Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics. 2022: 1758-1764.

[10] CHEN J, WANG Z, TIAN R, et al. Local additivity based data augmentation for semi-supervised NER[C]//Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing. 2020: 1241-1251.

[11] ZHANG H, CISSE M, DAUPHIN Y N, et al. mixup: Beyond empirical risk minimization[J]. arXiv preprint arXiv:1710.09412, 2017.

[12] KRUEGER K A, DAYAN P. Flexible shaping: How learning in small steps helps[J]. Cognition, 2009, 110(3): 380-394.

[13] DEVLIN J, CHANG M W, LEE K, et al. BERT: Pre-training of deep bidirectional transformers for language understanding[C]//Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics. 2019: 4171-4186.

[14] ANABY-TAVOR A, CARMELI B, GOLDBRAICH E, et al. Do not have enough data? Deep learning to the rescue![C]//Proceedings of the AAAI Conference on Artificial Intelligence: vol. 34: 05. 2020: 7383-7390.

[15] EDUNOV S, OTT M, AULI M, et al. Understanding back-translation at scale [C]//Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing. 2018: 489-500.

[16] BROWN T, MANN B, RYDER N, et al. Language models are few-shot learners[C]//Advances in Neural Information Processing Systems: vol. 33. 2020: 1877-1901.

[17] ZHAO Z, WALLACE E, FENG S, et al. Calibrate before use: Improving few-shot performance of language models[C]//International Conference on Machine Learning. 2021: 12697-12706.

[18] LU Y, BARTOLO M, MOORE A, et al. Fantastically ordered prompts and

where to find them: Overcoming few-shot prompt order sensitivity[C]//Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics. 2022: 8086-8098.

[19] LIU J, SHEN D, ZHANG Y, et al. What makes good in-context examples for GPT-3?[J]. arXiv preprint arXiv:2101.06804, 2021.

[20] MARTINO A, IANNELLI M, TRUONG C. Knowledge injection to counter large language model (LLM) hallucination[C]//European Semantic Web Conference. 2023: 182-185.

[21] XU Z, JAIN S, KANKANHALLI M. Hallucination is inevitable: An innate limitation of large language models[J]. arXiv preprint arXiv:2401.11817, 2024.

[22] WANG S, SUN X, LI X, et al. Gpt-ner: Named entity recognition via large language models[J]. arXiv preprint arXiv:2304.10428, 2023.

[23] WEI X, CUI X, CHENG N, et al. Zero-shot information extraction via chatting with chatgpt[J]. arXiv preprint arXiv:2302.10205, 2023.

[24] XIE T, LI Q, ZHANG J, et al. Empirical study of zero-shot ner with chatgpt [C]//Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing. 2023: 7935-7956.

[25] WEI J, WANG X, SCHUURMANS D, et al. Chain-of-thought prompting elicits reasoning in large language models[C]//Advances in Neural Information Processing Systems: vol. 35. 2022: 24824-24837.

[26] WANG Y, ZHANG Z, WANG R. Element-aware summarization with large language models: Expert-aligned evaluation and chain-of-thought method[C]//Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics. 2023: 8640-8665.

[27] MA X, LI J, ZHANG M. Chain of thought with explicit evidence reasoning for few-shot relation extraction[C]//Findings of the Association for Computational Linguistics: EMNLP 2023. 2023: 2334-2352.

[28] LEVY I, BOGIN B, BERANT J. Diverse demonstrations improve in-context

compositional generalization[C]//Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics. 2023: 1401-1422.

[29] WU Z, WANG Y, YE J, et al. Self-adaptive in-context learning: An information compression perspective for in-context example selection and ordering[C]//Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics. 2023: 1423-1436.

[30] NGUYEN T, WONG E. In-context example selection with influences[J]. arXiv preprint arXiv:2302.11042, 2023.

[31] LI X, QIU X. Finding support examples for in-context learning[C]//Findings of the Association for Computational Linguistics: EMNLP 2023. 2023: 6219-6235.

[32] WAN Z, CHENG F, MAO Z, et al. GPT-RE: In-context learning for relation extraction using large language models[C]//Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing. 2023: 3534-3547.

[33] VASWANI A, SHAZEER N, PARMAR N, et al. Attention is all you need[C]//Advances in Neural Information Processing Systems: vol. 30. 2017.

[34] LAMPLE G, BALLESTEROS M, SUBRAMANIAN S, et al. Neural architectures for named entity recognition[C]//Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics. 2016: 260-270.

[35] LUAN Y, WADDEN D, HE L, et al. A general framework for information extraction using dynamic span graphs[C]//Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics. 2019: 3036-3046.

[36] LI F, LIN Z, ZHANG M, et al. A span-based model for joint overlapped and discontinuous named entity recognition[C]//Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing. 2021: 4814-4828.

[37] YAN H, GUI T, DAI J, et al. A unified generative framework for various NER subtasks[C]//Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing. 2021: 5808-5822.

[38] SOHRAB M G, MIWA M. Deep exhaustive model for nested named entity recognition[C]//Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing. 2018: 2843-2849.

[39] SHEN Y, MA X, TAN Z, et al. Locate and label: A two-stage identifier for nested named entity recognition[C]//Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing. 2021: 2782-2794.

[40] LI X, FENG J, MENG Y, et al. A unified MRC framework for named entity recognition[C]//Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics. 2020: 5849-5859.

[41] 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 the 58th Annual Meeting of the Association for Computational Linguistics. 2020: 7871-7880.

[42] VINYALS O, FORTUNATO M, JAITLY N. Pointer networks[C]//Advances in Neural Information Processing Systems: vol. 28. 2015.

[43] CUI L, WU Y, LIU J, et al. Template-based named entity recognition using BART[C]//Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021. 2021: 1835-1845.

[44] LASKAR M T R, BARI M S, RAHMAN M, et al. A systematic study and comprehensive evaluation of chatgpt on benchmark datasets[C]//Findings of the Association for Computational Linguistics: ACL 2023. 2023: 431-469.

[45] MIKOLOV T, CHEN K, CORRADO G, et al. Efficient estimation of word representations in vector space[J]. arXiv preprint arXiv:1301.3781, 2013.

[46] PENNINGTON J, SOCHER R, MANNING C D. Glove: Global vectors for word representation[C]//Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing. 2014: 1532-1543.

[47] PETERS M E, NEUMANN M, IYYER M, et al. Deep contextualized word representations[C]//Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics. 2018: 2227-2237.

[48] HOCHREITER S, SCHMIDHUBER J. Long short-term memory[J]. Neural Computation, 1997, 9(8): 1735-1780.

[49] RADFORD A, NARASIMHAN K, SALIMANS T, et al. Improving language understanding by generative pre-training[J].

[50] KRIZHEVSKY A, SUTSKEVER I, HINTON G E. Imagenet classification with deep convolutional neural networks[C]//Advances in Neural Information Processing Systems: vol. 25. 2012.

[51] MIN B, ROSS H, SULEM E, et al. Recent advances in natural language processing via large pre-trained language models: A survey[J]. ACM Computing Surveys, 2023, 56(2): 1-40.

[52] RADFORD A, WU J, CHILD R, et al. Language models are unsupervised multitask learners[J].

[53] LIU Y, OTT M, GOYAL N, et al. RoBERTa: A robustly optimized bert pretraining approach[J]. arXiv preprint arXiv:1907.11692, 2019.

[54] 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.

[55] MINAEE S, MIKOLOV T, NIKZAD N, et al. Large language models: A survey [J]. arXiv preprint arXiv:2402.06196, 2024.

[56] WEI J, TAY Y, BOMMASANI R, et al. Emergent abilities of large language models[J]. arXiv preprint arXiv:2206.07682, 2022.

[57] ZHAO W X, ZHOU K, LI J, et al. A survey of large language models[J]. arXiv preprint arXiv:2303.18223, 2023.

[58] ACHIAM J, ADLER S, AGARWAL S, et al. GPT-4 technical report[J]. arXiv preprint arXiv:2303.08774, 2023.

[59] TOUVRON H, LAVRIL T, IZACARD G, et al. Llama: Open and efficient foundation language models[J]. arXiv preprint arXiv:2302.13971, 2023.

[60] TOUVRON H, MARTIN L, STONE K, et al. Llama 2: Open foundation and fine-tuned chat models[J]. arXiv preprint arXiv:2307.09288, 2023.

[61] TAORI R, GULRAJANI I, ZHANG T, et al. Alpaca: A strong, replicable instruction-following model[J]. Stanford Center for Research on Foundation Models. https://crfm. stanford. edu/2023/03/13/alpaca. html, 2023, 3(6): 7.

[62] CHIANG W L, LI Z, LIN Z, et al. Vicuna: An open-source chatbot impressing gpt-4 with 90%* chatgpt quality[EB/OL]. 2023. https://lmsys.org/blog/2023-03-30-vicuna/.

[63] LIU P, YUAN W, FU J, et al. Pre-train, prompt, and predict: A systematic survey of prompting methods in natural language processing[J]. ACM Computing Surveys, 2023, 55(9): 1-35.

[64] SCHICK T, SCHÜTZE H. Few-shot text generation with natural language instructions[C]//Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing. 2021: 390-402.

[65] SCHICK T, SCHÜTZE H. Exploiting cloze questions for few shot text classification and natural language inference[C]//Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics. 2021: 255-269.

[66] JIANG Z, XU F F, ARAKI J, et al. How can we know what language models know?[J]. Transactions of the Association for Computational Linguistics, 2020, 8: 423-438.

[67] WALLACE E, FENG S, KANDPAL N, et al. Universal adversarial triggers for

attacking and analyzing NLP[C]//Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing. 2019: 2153-2162.

[68] SHIN T, RAZEGHI Y, LOGAN IV R L, et al. Autoprompt: Eliciting knowledge from language models with automatically generated prompts[C]//Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing. 2020: 4222-4235.

[69] YUAN W, NEUBIG G, LIU P. Bartscore: Evaluating generated text as text generation[C]//Advances in Neural Information Processing Systems: vol. 34. 2021: 27263-27277.

[70] GAO T, FISCH A, CHEN D. Making pre-trained language models better few-shot learners[C]//Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing. 2021: 3816-3830.

[71] BEN-DAVID E, OVED N, REICHART R. PADA: Example-based prompt Learning for on-the-fly adaptation to unseen domains[J]. Transactions of the Association for Computational Linguistics, 2022, 10: 414-433.

[72] LIU X, ZHENG Y, DU Z, et al. GPT understands, too[J]. AI Open, 2023.

[73] LI X L, LIANG P. Prefix-tuning: Optimizing continuous prompts for generation [C]//Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing. 2021: 4582-4597.

[74] LESTER B, AL-RFOU R, CONSTANT N. The power of scale for parameter-efficient prompt tuning[C]//Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing. 2021: 3045-3059.

[75] PASSIGAN P, YOHANNES K, PEREIRA J. Continuous prompt generation from linear combination of discrete prompt embeddings[J]. arXiv preprint arXiv:2312.10323, 2023.

[76] QIN G, EISNER J. Learning how to ask: Querying LMs with mixtures of soft prompts[C]//Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics. 2021: 5203-5212.

[77] ZHONG Z, FRIEDMAN D, CHEN D. Factual probing is [MASK]: Learning vs. learning to recall[C]//Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics. 2021: 5017-5033.

[78] KAPLAN J, MCCANDLISH S, HENIGHAN T, et al. Scaling laws for neural language models[J]. arXiv preprint arXiv:2001.08361, 2020.

[79] HONOVICH O, SHAHAM U, BOWMAN S, et al. Instruction induction: From few examples to natural language task descriptions[C]//Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics. 2023: 1935-1952.

[80] ZHOU Y, MURESANU A I, HAN Z, et al. Large language models are human-level prompt engineers[J]. arXiv preprint arXiv:2211.01910, 2022.

[81] PRYZANT R, ITER D, LI J, et al. Automatic prompt optimization with" gradient descent" and beam search[C]//Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing. 2023: 7957-7968.

[82] COBBE K, KOSARAJU V, BAVARIAN M, et al. Training verifiers to solve math word problems[J]. arXiv preprint arXiv:2110.14168, 2021.

[83] MIHAYLOV T, CLARK P, KHOT T, et al. Can a suit of armor conduct electricity? A new dataset for open book question answering[C]//Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing. 2018: 2381-2391.

[84] KOJIMA T, GU S S, REID M, et al. Large language models are zero-shot reasoners[C]//Advances in Neural Information Processing Systems: vol. 35. 2022: 22199-22213.

[85] ZHANG Z, ZHANG A, LI M, et al. Automatic chain of thought prompting in

large language models[J]. arXiv preprint arXiv:2210.03493, 2022.

[86] CHEN W, MA X, WANG X, et al. Program of thoughts prompting: Disentangling computation from reasoning for numerical reasoning tasks[J]. arXiv preprint arXiv:2211.12588, 2022.

[87] YAO S, YU D, ZHAO J, et al. Tree of thoughts: Deliberate problem solving with large language models[C]//Advances in Neural Information Processing Systems: vol. 36. 2024.

[88] LONG J. Large language model guided tree-of-thought[J]. arXiv preprint arXiv:2305.08291, 2023.

[89] BESTA M, BLACH N, KUBICEK A, et al. Graph of thoughts: Solving elaborate problems with large language models[C]//Proceedings of the AAAI Conference on Artificial Intelligence: vol. 38: 16. 2024: 17682-17690.

[90] LEI B, LIAO C, DING C, et al. Boosting logical reasoning in large language models through a new framework: The graph of thought[J]. arXiv preprint arXiv:2308.08614, 2023.

[91] LING Z, FANG Y, LI X, et al. Deductive verification of chain-of-thought reasoning[C]//Advances in Neural Information Processing Systems: vol. 36. 2024.

[92] WANG X, WEI J, SCHUURMANS D, et al. Self-consistency improves chain of thought reasoning in language models[J]. arXiv preprint arXiv:2203.11171, 2022.

[93] LI Y, LIN Z, ZHANG S, et al. Making language models better reasoners with step-aware verifier[C]//Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics. 2023: 5315-5333.

[94] WENG Y, ZHU M, XIA F, et al. Large language models are better reasoners with self-verification[C]//Findings of the Association for Computational Linguistics: EMNLP 2023. 2023: 2550-2575.

[95] ZHOU D, SCHÄRLI N, HOU L, et al. Least-to-most prompting enables complex reasoning in large language models[J]. arXiv preprint arXiv:2205.10625, 2022.

[96] DUA D, GUPTA S, SINGH S, et al. Successive prompting for decomposing complex questions[C]//Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing. 2022: 1251-1265.

[97] HE Z, LIANG T, JIAO W, et al. Exploring human-like translation strategy with large language models[J]. Transactions of the Association for Computational Linguistics, 2024, 12: 229-246.

[98] ZHANG Z, ZHANG A, LI M, et al. Multimodal chain-of-thought reasoning in language models[J]. arXiv preprint arXiv:2302.00923, 2023.

[99] DIAO S, WANG P, LIN Y, et al. Active prompting with chain-of-thought for large language models[J]. arXiv preprint arXiv:2302.12246, 2023.

[100] SU H, KASAI J, WU C H, et al. Selective annotation makes language models better few-shot learners[J]. arXiv preprint arXiv:2209.01975, 2022.

[101] YE X, IYER S, CELIKYILMAZ A, et al. Complementary explanations for effective in-context learning[C]//Findings of the Association for Computational Linguistics: ACL 2023. 2023: 4469-4484.

[102] MA H, ZHANG C, BIAN Y, et al. Fairness-guided few-shot prompting for large language models[C]//Advances in Neural Information Processing Systems: vol. 36. 2024.

[103] SUN C, SHRIVASTAVA A, SINGH S, et al. Revisiting unreasonable effectiveness of data in deep learning era[C]//Proceedings of the IEEE International Conference on Computer Vision. 2017: 843-852.

[104] ALOM M Z, TAHA T M, YAKOPCIC C, et al. A state-of-the-art survey on deep learning theory and architectures[J]. Electronics, 2019, 8(3): 292.

[105] LEE S, LEE H, YOON S. Adversarial vertex mixup: Toward better adversarially robust generalization[C]//Proceedings of the IEEE/CVF Conference on

Computer Vision and Pattern Recognition. 2020: 272-281.

[106] SANG E T K, DE MEULDER F. Introduction to the CoNLL-2003 shared task: Language-independent named entity recognition[C]//Proceedings of the Seventh Conference on Natural Language Learning at HLT-NAACL 2003. 2003: 142-147.

[107] LIU J, PASUPAT P, CYPHERS S, et al. Asgard: A portable architecture for multilingual dialogue systems[C]//2013 IEEE International Conference on Acoustics, Speech and Signal Processing. 2013: 8386-8390.

[108] PAN X, ZHANG B, MAY J, et al. Cross-lingual name tagging and linking for 282 languages[C]//Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics. 2017: 1946-1958.

[109] WANG Y, MUKHERJEE S, CHU H, et al. Meta self-training for few-shot neural sequence labeling[C]//Proceedings of the 27th ACM SIGKDD Conference on Knowledge Discovery & Data Mining. 2021: 1737-1747.

[110] CHU Z, CHEN J, CHEN Q, et al. A survey of chain of thought reasoning: Advances, frontiers and future[J]. arXiv preprint arXiv:2309.15402, 2023.

[111] EgoAlpha. Prompt engineering[EB/OL]. 2023. https://github.com/EgoAlpha/prompt-in-context-learning/blob/main/PromptEngineering.md.

[112] REIMERS N, GUREVYCH I. Sentence-BERT: Sentence embeddings using siamese bert-networks[C]//Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing. 2019: 3982-3992.

[113] LIU Z, XU Y, YU T, et al. CrossNER: Evaluating cross-domain named entity recognition[C]//Proceedings of the AAAI Conference on Artificial Intelligence: vol. 35: 15. 2021: 13452-13460.

致　　谢

三年的研究生时光匆匆而逝，这篇毕业论文不仅是我学术探索的成果，更是我人生旅程中一个重要阶段的总结。在这段宝贵的时光里，我收获的不仅是知识，还有成长与感悟。在此，我要向所有给予我帮助和支持的人表达我最深切的感谢。

首先，我要感谢我的导师申富饶教授。申老师不仅以他深厚的学识和严谨的治学态度为我指引方向，更以他宽厚的人格魅力和崇高的师德深深影响着我。在我遇到研究瓶颈时，申老师总是耐心倾听，细致指导，帮助我一一克服。在我信心动摇或情绪低落时，申老师总能用他的智慧和经验给予我鼓励和安慰，让我重新找到前进的方向。申老师的言传身教，将是我未来人生道路上的宝贵财富。

其次，我还要感谢 RINC 实验室的每一位成员。这是一个充满活力和智慧的团队。在这个温暖的大家庭中，我得到了无数的帮助和启发，我们共同经历了一段充满挑战和收获的旅程，一起踩坑，也一起进步和成长。

此外，我还要特别感谢我的家人和朋友们。他们是我坚强的后盾，始终给予我无条件的爱与支持。在我疲惫或沮丧时，是他们的陪伴和鼓励，让我重新获得勇气和信心；在我快乐和成功时，是他们与我一同分享喜悦，为我感到骄傲。

在未来的道路上，我将带着这份感激，继续前行，不断探索。再次感谢所有给予我帮助和支持的人，是你们让这段旅程变得如此丰富和有意义。

我们下段旅程再见！

附录 A　上下文学习中使用的提示

本节给出了4.2.4节的数据增强模块中所用提示的参考。由于篇幅的限制，这里仅给出任务描述、输入查询以及输出结果的关键部分，省略了一些细节以及演示示例部分。图A-1和图A-2分别表示引导大模型进行数据增强以及进行自一致过滤的提示过程。

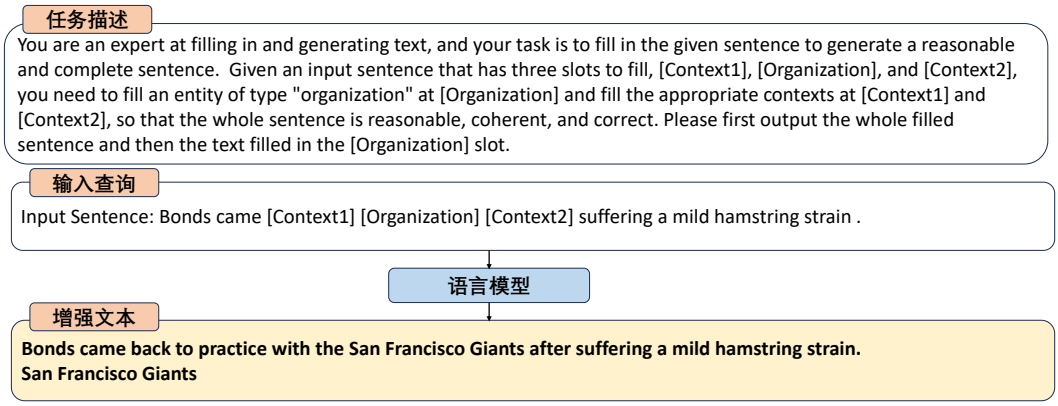


图 A-1　数据增强使用的提示

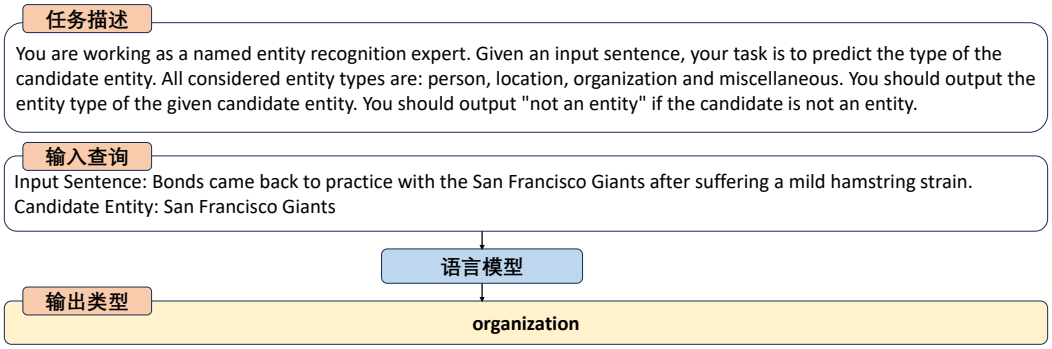


图 A-2　自一致过滤使用的提示

简历与科研成果

基本信息

宋斯涵，女，汉族，1999 年 11 月出生，河南省新乡人。

教育背景

2021 年 9 月—2024 年 6 月　南京大学计算机科学与技术系　硕士

2017 年 9 月—2021 年 6 月　南京大学计算机科学与技术系　本科

攻读硕士学位期间完成的学术成果

1. **Sihan Song**, Furao Shen, Jian Zhao. RoPDA: Robust Prompt-Based Data Augmentation for Low-Resource Named Entity Recognition[C]//Proceedings of the AAAI Conference on Artificial Intelligence. 2024, 38(17): 19017-19025.

攻读硕士学位期间参与的科研课题

1. 科技部重大项目“基于神经可塑性的脉冲网络高效学习机制与类脑智能系统”（项目编号 2021ZD0201300，课题年限 2021 年 9 月—2024 年 6 月）。
2. 国家自然科学基金项目“面向增量式无监督学习的新型神经网络研究”（项目编号 62276127，课题年限 2023 年 1 月—2024 年 4 月）。