生物医学命名实体识别的有效多任务学习

João Ruano, Gonçalo M. Correia, Leonor Barreiros and Afonso Mendes
Priberam Labs, Alameda D. Afonso Henriques, 41, 2ž, 1000-123 Lisboa, Portugal
{ joao.ruano,goncalo.correia,leonor.barreiros,amm } @priberam.pt

Abstract

生物医学命名实体识别由于生物医学术语的复杂性和数据集之间注释的不一致性，面临重大挑战。本文介绍了 SRU-NER（基于槽的递归单元 NER），这是一种新颖的方法，旨在处理嵌套的命名实体，同时通过有效的多任务学习策略整合多个数据集。SRU-NER 通过动态调整损失计算来减轻注释差距，从而避免对给定数据集中不存在的实体类型的预测进行惩罚。[1] 通过广泛的实验，包括跨语料库评估和对模型预测的人为评估，SRU-NER 在生物医学和通用域的 NER 任务中取得了有竞争力的性能，同时提高了跨域泛化能力。

1 引言

命名实体识别（NER）是多个自然语言处理管道中的一个关键步骤，例如信息抽取、信息检索、机器翻译和问答系统（Sharma et al., 2022）。给定非结构化文本，NER 的任务是根据感兴趣的类别识别和分类文本片段。这些类别根据下游应用而定义，可以从通用类别（如人物、地点、组织）到特定领域的生物医学实体（如基因、疾病、化学物质）不等。

特别是，由于生物医学命名法的复杂性，生物医学命名实体识别（BioNER）具有挑战性。从形态上看，这些实体可以包含希腊字母、数字、标点（如 α -微管蛋白，IL-6），形式变体（如抑制剂 vs. 抑制性），以及复合术语（如肿瘤坏死因子- vs. TNF- α ）。从语义上看，多义性（例如，p53 可以指代基因、蛋白质或病况）增加了歧义。这些挑战使得人工标注成本高昂，导致 BioNER 数据集规模较小，且通常仅关注有限数量的实体类型（Greenberg et al., 2018）。

在构建 BioNER 模型时解决数据稀缺问题的一种方法是利用多个数据集，每个数据集都注释了特定子集的实体。然而，仅仅在所有可用数据集的联合上训练单一模型假设每种实体类型在所有训练实例中是一致注释的，事实并非如此。这导致了假阴性的高度流行，因为在一个数据集中标记的实体可能在另一个数据集中被完全忽略。另一方面，为每个数据集训练单独的模型未能利用跨数据集的共享统计模式，并在推理时引入了解决预测冲突的挑战（Greenberg et al., 2018）。因此，一个有效的策略必须在从多个来源学习的同时平衡考虑缺失的注释和标记方案的不一致。

我们的贡献包括以下三方面：(i) 我们引入了 SRU-NER（基于槽的递归单元命名实体识别），这是一种能够通过生成动作序列解决嵌套命名实体识别的模型；(ii) 我们提出了一种有效的多任务训练策略，以处理在单一模型中利用多个命名实体识别数据集的复杂挑战；以及 (iii) 我们展示了 SRU-NER 如何通过多项实验来处理单一共享网络上的多个数据集，包括跨语料库评估和在不相交实体集的语料库上的人工评估。

2 相关工作

命名实体识别在过去几十年中有了显著的发展。早期系统依赖于基于规则的方法，这些方法可解释但缺乏灵活性。机器学习的引入使得方法更加灵活，随后深度学习技术进一步增强了这种灵活性，能够捕捉复杂的语言模式。最近，基于 Transformer 的架构树立了新的基准，推动了 NER 性能上的重大进步。在 CoNLL-2003 数据集中，作为 NER 任务的一个基准，性能有了显著提高，F1 得分已超过 94 %。同样的现象也出现在 GENIA 语料库中，这是一个嵌套的生物 NER 数据集，测试 F1 得分超过 80 % 。

为了解决 BioNER 数据集的激增问题，几项研究转向了多任务学习（MTL；Park et al., 2024）。传统的深度学习 NER 模型在单一数据集上训练，被称为单任务模型，因为它们专注于识别其训练数据中标注的特定实体类型的提及范围。单任务模型在域外设置上的表现通常较差。相比之下，MTL 框架利用多个数据集，每个数据集对应一个不同的任务，使模型能够从

[1] 代码公开可在 https://github.com/Priberam/sru-ner 获取。

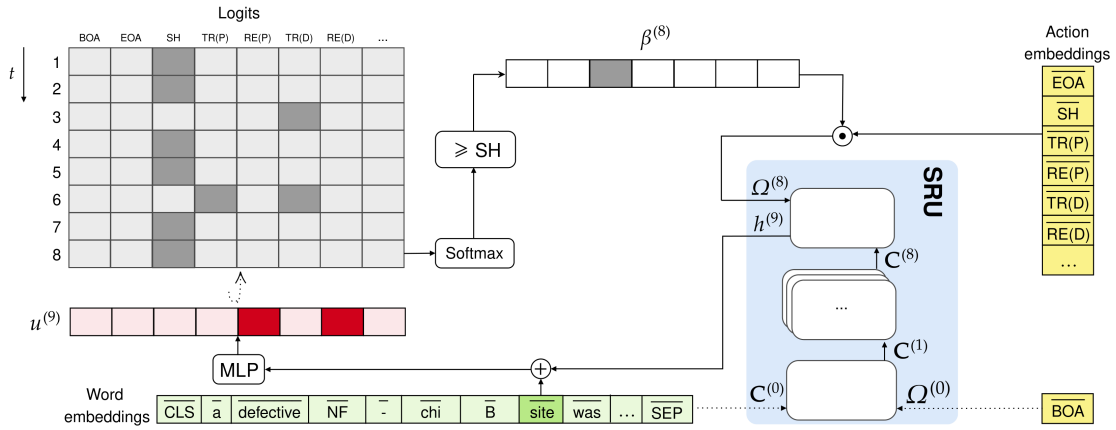

Figure 1: 在时间步 $t = 9$ 处，针对在 3.1 节中给出的句子的动作选择过程。金标准嵌套提及为类型为 DNA (D) 的"NF - chi B site"、"chi B"，以及类型为 Protein (P) 的"NF - chi B"。为了计算 $u^{(9)}$，模型利用了先前时间步的对数、动作嵌入和词嵌入。

不同来源学习。基本前提是，不同的数据集共享可以共同利用的信息，以鼓励学习更广泛的表示，从而提高模型的鲁棒性 (Mehmood et al., 2019; Li et al., 2022) 。

MTL 学习框架可以根据跨任务共享的模块分为两类：(i) 共享编码层而维护任务特定解码层的框架 (Crichton et al., 2017; Wang et al., 2018; Khan et al., 2020) , 以及 (ii) 共享所有层的框架 (Greenberg et al., 2018; Huang et al., 2019; Banerjee et al., 2021; Luo et al., 2023; Moscato et al., 2023) 。SRU-NER 类似于类型 (ii) 的模型，它在所有任务中共享其解码层。通常，这些模型在误报方面存在自然问题，因为统一的解码器可能难以区分任务特定的实体边界和标签，从而导致遗漏有效实体。我们的方法通过一种有效的多任务学习策略避免了这一问题。

3 有效的命名实体识别多任务学习

所提出的模型 SRU-NER 以类似于基于转换的解析器 (Dyer et al., 2015; Marinho et al., 2019) 的方式解决嵌套命名实体识别任务。给定一个单词序列 $S = [w_1, w_2, \ldots, w_N]$ ，模型生成一系列动作。在每个时间步，动作的选择依赖于句子的单词和先前选择的动作。在解析过程结束时，完整的动作序列被解码为提及。

3.1 动作编码

考虑到系统经过训练来识别属于 $\mathbb{E} = \{e_1, e_2, \ldots, e_M\}$ 的实体类型的提及。令 $\mathcal{A}_{\mathbb{E}}$ 代表系统的 $2M + 2$ 个可能动作：两个特殊的标记 (SH 和 EOA)，每个实体类型 e_i 对应有一对动作，分别表示为 TR(e_i) 和 RE(e_i) 。TR(e_i) ，是“转移到实体 e_i ”的缩写，表示开始提及类型为 e_i 的实体；这就是说，此动作开启了一个类型为 e_i 的提及。RE(e_i) ，是“缩减实体 e_i ”的缩写，表示最近开启的 e_i 类型提及的结束；这就是说，这个动作关闭了一个提及。SH ，是“移动”的缩写，意味着输入指针应移至下一个标记；因此，句子中的每个词都有一个 SH 。最后，EOA 是结束动作。

这些动作通过选择的顺序有效地编码了嵌套提及。如果一个类型为 e_k 的提及从单词 w_i 开始，并在单词 w_j 结束，TR(e_k) 出现在代表第 i 个单词的 SH 之前，而 RE(e_j) 出现在代表第 k 个单词的 SH 之后；如果两个提及从同一个单词开始，最长提及的 TR() 首先出现；反之，如果两个提及在同一个单词结束，最短提及的 RE() 首先出现。请考虑 GENIA 数据集中以下句子 (Kim et al., 2003) :

Protein DNA

a defective NF - chi B site was completely ...

这个句子包含嵌套的指称，例如，类型为蛋白质的指称“NF - chi B”包含在类型为 DNA 的指称“NF - chi B site”中。句子的动作编码及其指称为：SH → SH → TR(DNA) → TR($Protein$) → SH → SH → TR(DNA) → SH → SH → RE(DNA) → RE($Protein$) → SH → RE(DNA) → SH → SH → ... → EOA 。

3.2 总体架构

使用先前的符号，假设需要检测 $\mathbb{E}$ 在句子 S 中的提及。模型由三个连续的步骤组成：将 S 编码为一个稠密的上下文嵌入矩阵 $\boldsymbol{S}$ ，迭代式动作生成过程，以及将选定的动作解码为句子中存在的提及。

在第一步中，S 通过一个类似 BERT 的编码器，生成一个上下文嵌入矩阵。对于每个单词 w_i，通过对其子词的嵌入进行最大池化，获得其密集嵌入，记为 $\overline{w_i}$。这样，编码后的句子 $\boldsymbol{S}$ 是一个大小为 $(N+2, d_{\text{enc}})$ 的张量，其中 d_{enc} 是编码器嵌入维度，$\overline{\text{CLS}}$（分别为 $\overline{\text{SEP}}$）是编码器的分类（分别为分隔）标记的嵌入。

在给定 $\boldsymbol{S}$ 的情况下，模型进入一个迭代的动作选择过程，在每个时间步骤 t 中，计算每个在 $\mathcal{A}_{\mathbb{E}}$ 中可能动作的对数概率。[2] 图 1 展示了该循环一个时间步骤的示意图。

更具体地，定义 $u_{a_i}^{(t)}$ 为动作 $a_i \in \mathcal{A}_{\mathbb{E}}$ 在时间步 t 的 logit 值。假设系统已经计算了前 $T \geq 1$ 个时间步的这些值，因此即将为时间步 $t = T+1$ 计算这些值。根据上一节，SH 动作对应于在句子 S 中推进一个标记。因此，定义

$$p^{(t)} = \sum_{t_0 \leq t} \mathbb{1}\left(\arg\max_{a_i \in \mathcal{A}_{\mathbb{E}}} \left(u_{a_i}^{(t_0)}\right) = \text{SH}\right), \quad (1)$$

，其中 $\mathbb{1}$ 代表指示函数。因此，$p^{(t)}$ 是指在之前时间步 t 中已经解析的标记数，对于 $1 \leq t \leq T$。最后，为每个 $1 \leq t \leq T$ 定义，

$$\Omega^{(t)} = \sum_{a_i \in \mathcal{A}_{\mathbb{E}}} \beta_{a_i}^{(t)} \, \overline{a_i}, \quad (2)$$

，其中 $\overline{a_i}$ 是大小为 d_{enc} 的训练嵌入

$$\beta_{a_i}^{(t)} = \begin{cases} u_{a_i}^{(t)} & \text{if } u_{a_i}^{(t)} \geq u_{\text{SH}}^{(t)} \\ 0 & \text{otherwise} \end{cases}$$

换句话说，$\Omega^{(t)}$ 是在时间步骤 t 所选择的动作的加权嵌入，其中低于 SH 的 logit 的动作被排除。

令 $\boldsymbol{u}^{(T+1)}$ 为 $u_{a_i}^{(T+1)}$ 在 $a_i \in \mathcal{A}_{\mathbb{E}}$ 上的 logits 向量。它们被计算为

$$\boldsymbol{u}^{(T+1)} = \text{MLP}\left(f\left(p^{(T)}, \Omega^{(T)}\right)\right), \quad (3)$$

其中 MLP 由一个 dropout 层、一个全连接层、一个 tanh 激活以及一个线性层组成，该线性层的输出节点对应于 $\mathcal{A}_{\mathbb{E}}$ 中的每个动作。这个 MLP 的输入是

$$f\left(p^{(T)}, \Omega^{(T)}\right) = \boldsymbol{S}_{p^{(T)}+1} \oplus \text{SRU}\left(\Omega^{(T)}, p^{(T)}\right),$$

[2] 与基于标记的标记方法不同，总的时间步数不是事先确定的，尽管始终下界为 N，即 S 中的单词数量。

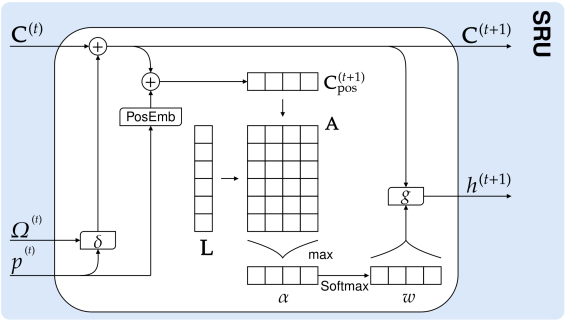

Figure 2: 在时间步 t 的 SRU 单元。其内部状态根据其当前状态 $\boldsymbol{C}^{(t)}$ 和加权动作嵌入 $\Omega^{(t)}$ 进行更新。该有状态函数还利用一组潜在表示。通过将注意力机制应用于更新后的状态，它生成一个输出嵌入 $h^{(t+1)}$。

，即下一个标记的嵌入 $\boldsymbol{S}_{p^{(T)}+1}$ 与“已处理动作记忆”的最后状态的嵌入的串联。这种记忆保存着动作历史，并在每次调用时通过利用一组内部潜在表示来计算加权嵌入。该模块被称为基于插槽的递归单元（SRU），在第 3.3 节中进行描述。

为了进行第一次预测 $\boldsymbol{u}^{(1)}$，系统通过设置 $p^{(0)} = 0$ 和 $\Omega^{(0)}$ 为另一个尺寸为 d_{enc} 的训练嵌入来初始化，该嵌入记作 $\overline{\text{BOA}}$。[3] 当达到时间步 $t = T_{\text{final}}$ 时，动作生成循环终止，使得

$$\text{Sigmoid}\left(u_{\text{EOA}}^{(T_{\text{final}})}\right) > 0.5\,. \quad (4)$$

在动作生成循环结束时，来自所有时间步的输出 logits 通过一个 sigmoid 函数。这为 $\mathcal{A}_{\mathbb{E}}$ 中的每个动作产生了一组独立的概率分数，从中提取提及跨度。解码模块为 $\mathbb{E}$ 中的每种实体类型维护了单独的开放跨度堆栈，允许不同类型的跨度重叠。

解码过程遍历概率得分列表，直到达到最高得分动作为 EOA [4] 的时间步。在到达该时间步之前，解码器遵循两个规则：(i) 如果最高得分动作是 SH，则递增计算已解析单词数的指针；(ii) 如果最高得分动作为 TR() 或 RE()，则更新实体提及栈。在后一种情况下，仅考虑概率得分高于 0.5 的动作。转移动作开启新的跨度，而规约动作关闭最近的相应实体类型的跨度，如在第 3.1 节中讨论的那样。

3.3 基于槽的循环单元

基于槽的循环单元（SRU）是一个有状态的函数，在每个时间步，它接受一对输入，更新其内部状态，并生成一个输出嵌入。

[3] 在本文中，采用从零开始的索引表示法用于张量，因此 $\boldsymbol{S}_{p_0+1} = \overline{w_1}$。

[4] 尽管这一停止条件与方程（4）中存在的动作生成过程不同，但实验证明它能够提供更好的结果。

在每个时间步 t ，SRU 根据以下内容更新其内部状态

$$\boldsymbol{C}^{(t+1)} = m\left(\boldsymbol{C}^{(t)}, \Omega^{(t)}, p^{(t)}\right) ,$$

，其中 $\boldsymbol{C}^{(t)} \in \mathbb{R}^{Q\times d}$ 是 SRU 的内部状态矩阵，$\Omega^{(t)} \in \mathbb{R}^{d}$ 是输入向量，$p^{(t)} \in \{0, 1, \ldots, Q-1\}$ 是输入整数。它还通过以下方式产生输出嵌入 $h^{(t+1)} \in \mathbb{R}^{d}$

$$h^{(t+1)} = g\left(\boldsymbol{C}^{(t+1)}, p^{(t)}\right) .$$

的示意图如图 2 所示。Q 和 d 分别指代内部状态矩阵中的行（或槽）数量以及输入和输出嵌入的隐藏维度。

函数通过将输入向量 $\Omega^{(t)}$ 加到 $p^{(t)}$ 行来更新 $\boldsymbol{C}^{(t)}$ ，即。

$$m\left(\boldsymbol{C}^{(t)}, \Omega^{(t)}, p^{(t)}\right) := \boldsymbol{C}^{(t)} + \delta_{p^{(t)}} \left(\Omega^{(t)}\right)^{T}$$

，其中 $\delta_{p^{(t)}} \in \mathbb{R}^{Q}$ 是一个在其 $p^{(t)}$ -坐标上的 1 的 one-hot 向量。

输出嵌入 $h^{(t)} \in \mathbb{R}^{d}$ 通过函数 g 获得，该函数定义为

$$g\left(\boldsymbol{C}^{(t+1)}, p^{(t)}\right) := \boldsymbol{w}^{T} \left(\boldsymbol{C}^{(t+1)}\, \boldsymbol{D}_1\right)$$

，其中 $\boldsymbol{D}_1$ 是 d 大小的训练过的对角矩阵，$\boldsymbol{w} \in \mathbb{R}^{Q}$ 是通过受 Ganea and Hofmann, 2017 启发的注意力机制计算出的权重，具体如下。首先，通过添加位置信息来增强 $\boldsymbol{C}^{(t+1)}$ 。

$$\boldsymbol{C}_{\text{pos}}^{(t+1)} = \alpha\, \boldsymbol{C}^{(t+1)} + \text{Dropout}\left(\boldsymbol{P}\left(p^{(t)}\right)\right) \quad (5)$$

其中 α 是一个经过训练的缩放参数，$\boldsymbol{P}\left(p^{(t)}\right) \in \mathbb{R}^{Q\times d}$ 是位置嵌入。[5] 接下来，使用一组大小为 d 的 J 训练得到的潜在嵌入来计算 $\boldsymbol{C}^{(t+1)}$ 中每一行的注意力分数。定义 $\boldsymbol{L} \in \mathbb{R}^{J\times d}$ 为潜在嵌入的矩阵，通过以下方式计算注意力分数矩阵

$$\boldsymbol{A} = \text{Dropout}(\boldsymbol{L})\, \boldsymbol{D}_2 \left(\boldsymbol{C}_{\text{pos}}^{(t+1)}\right)^{T} ,$$

其中 $\boldsymbol{D}_2$ 为尺寸为 d 的训练好的对角矩阵。通过为 $q \in \{0, 1, \ldots, Q-1\}$ 设置 $\boldsymbol{s} = \max_j(A_{jq})$ 来获得每个槽位的注意力分数。最后，这些分数 $\boldsymbol{s}$ 通过 softmax 进行归一化以得到权重 $w \in \mathbb{R}^{Q}$ 。

SRU 模块在每个动作生成的时间步中用于计算一个嵌入，该嵌入用于建模“已处理动作内存”栈的当前状态。对于每个时间步 t ，输入整数 $p^{(t)}$ 由方程 (1) 定义，而输入向量 $\Omega^{(t)}$ 由方程 (2) 定义。此外，d 被设定为编码器嵌入维度 d_{enc} ，槽的数量为 $Q = N + 2$ ，潜变量的数量 J 为 $|\mathcal{A}_{\mathbb{E}}| = 2M + 2$ 的整数倍 [6] 。内部状态矩阵通过设定 $\boldsymbol{C}^{(0)} = \boldsymbol{S}$ 来初始化。在考虑这种初始化选择的情况下，并参考方程 (3)，在计算 $h^{(T+1)} = \text{SRU}\left(\Omega^{(T)}, p^{(T)}\right)$ 时，更新后的内部状态矩阵 $\boldsymbol{C}^{(T+1)}$ 的所有槽都填充了编码句子的嵌入 $\boldsymbol{S}$ 。此外，先前选择的动作的历史记录存在于 $\boldsymbol{C}^{(T+1)}$ 中，因为在之前时间步 $0 \le t \le T$ 每次调用 SRU 模块时，方程 (2) 中的加权动作嵌入 $\Omega^{(t)}$ 被加到了由 $p^{(t)}$ 指向的槽上。

假设该模型在一个包含 K 个数据集 $\mathcal{D} = \{D_i\}_{i=1}^{K}$ 的集合上进行训练，其中每个数据集 D_i 都标注了实体类型的区间。为了考虑标签方案的差异，在训练过程中，不同数据集的实体类型总是被认为是不同的。因此，模型被训练以在非交集联合集 $\widehat{\mathbb{E}} = \bigsqcup_{i=1}^{K} \mathbb{E}_i$ 中识别实体类型的区间。

模型的训练目标是最小化一个批次中样本的平均损失。每个批次是通过从 $\mathcal{D}$ 中随机选择样本构建的。为了确保所有数据集的贡献平衡，从给定数据集中选择样本的概率与该数据集中句子的总数成反比。每个 epoch 的总样本数是 $\mathcal{D}$ 数据集中句子平均数量。

令 S 为批次中的一个句子，来自数据集 D_i ，因此用实体类型 $\mathbb{E}_i$ 的黄金跨度进行标注。动作生成循环的输出是一个矩阵

$$\boldsymbol{U} = \left(u_{a_i}^{(t)}\right)_{t=1,\ldots,T_{\text{final}};\, a_i\in\mathcal{A}_{\mathbb{E}}} ,$$

每行 $u_{*}^{(t)}$ 包含关于与不相交并集 $\widehat{\mathbb{E}}$ 相关的所有动作 $\mathcal{A}_{\widehat{\mathbb{E}}}$ 的时间步 t 的对数。[7] 为了计算 $\boldsymbol{U}$ 的损失值，将实施以下约束条件：

i) 一方面，模型应该由于未能预测出与实体类型 $\mathbb{E}_i$ 的金标准跨度相对应的 `TR()` 和 `RE()` 动作而被惩罚，而这些实体类型的金标准跨度中标注了 S ；但是

[5] 这些位置嵌入是相对的，意味着 $\boldsymbol{P}\left(p^{(t)}\right)$ 的每一行是根据其与索引为 $p^{(t)}$ 的行的距离，从一个训练好的嵌入表中选择的。

[6] 对于所进行的实验，它被设置为 2 或 10 （参见附录 B 中的表格 12 ）。

[7] 在推理时，当 `EOA` 动作的概率超过阈值时，动作生成过程停止，如在 3.2 节所述。然而，在训练过程中，为了保证所有的金标准动作都得到考虑，循环只有在所有标记被解析（即移入）后才会停止。

Dataset	SRU-NER		Wang et al., 2018	Huang et al., 2019	Khan et al., 2020	Moscato et al., 2023
	Merged	Disjoint				
BC2GM	78.80	83.95	80.74 *	79.1	83.01 *	84.84
BC4CHEMD	90.42	92.05	89.37 *	87.3	—	—
BC5CDR	89.37	90.26	88.78 *	—	89.50 *	◇
JNLPBA	72.15	76.00	73.52 *	83.8	72.89 *	—
Linnaeus	88.82		—	83.9	—	—
NCBI Disease	87.32	88.71	86.14 *	84.0	88.10 *	89.20
Average	84.48	86.63				

Table 1: 一些多任务模型在六个生物医学数据集的子集上训练的 Micro-F1 分数。对于 SRU-NER，分数依据两种评估场景（合并和不相交）进行报告，具体如章节 3.5 所述。最佳分数为加粗字体，次优分数为 underlined 。符号说明：
—：数据集在训练中缺席；
*：模型在语料库的训练和开发拆分上进行训练；
◇：模型仅使用 BC5CDR 的 “Chemical” 注释进行训练，获得 F1 为 93.95；对于同一标签，SRU-NER 在不相交评估中获得 F1 为 93.77，合并评估中获得 F1 为 93.18。

ii) 另一方面，模型不应因在 $\widehat{\mathbb{E}} \setminus \mathbb{E}_i$ 中预测 TR() 和 RE() 实体类型的动作而受到惩罚，因为这些动作在 S 中没有被标注。

在实际操作中，该策略的应用如下。与句子 S 的黄金注释相对应的动作列表（如第 3.1 节详细构建的，并考虑了不相交的实体类型集合 $\widehat{\mathbb{E}}$ ）被扩展为一个矩阵 $\boldsymbol{G} = \left(G_{a_i}^{(t)}\right) \in \mathbb{R}^{T_{\text{initial}} \times |\mathcal{A}_{\widehat{\mathbb{E}}}|}$ ，其中每行 $G_*^{(t)}$ 是一个多热向量，表示一个特定的时间步长 t ，在对应于黄金动作的列中填充 1 。这种转换方式确保了 SH 和 EOA 动作总是占据不同的时间步骤，但不同实体类型的 TR() 和 RE() 动作可以在同一个时间步骤共存。然后，在动作生成循环中，通过结合模型对来自其他数据集的 TR() 和 RE() 动作决策的概率来修改 $\boldsymbol{G}$ 。更具体地说，在循环的时间步骤 t ，对于 $a_i \in \mathcal{A}_{\widehat{\mathbb{E}}} \setminus \mathcal{A}_{\mathbb{E}_i}$ ，$G_{a_i}^{(t)}$ 被设置为等于 $\sigma\left(u_{a_i}^{(t)}\right)$ ，其中 σ 是 sigmoid 函数。此外，当在某些 $a_i \in \mathcal{A}_{\widehat{\mathbb{E}}} \setminus \mathcal{A}_{\mathbb{E}_i}$ 情况下，$G_{\text{SH}}^{(t)} = 1$ 和 $u_{a_i}^{(t)} > u_{\text{SH}}^{(t)}$ ，即模型尝试打开/关闭来自其他数据集 D_j （$j \neq i$ ）的实体类型的新跨度时，值 $G_{\text{SH}}^{(t)}$ 被更改为 $\sigma\left(u_{\text{SH}}^{(t)}\right)$ 。在这种情况下，会在 $\boldsymbol{G}$ 的 $G_*^{(t)}$ 之后插入一个单热向量，以便在下一时间步长 $t+1$ ，所有 $a_i \in \widehat{\mathbb{E}} \setminus \{\text{SH}\}$ 的 $G_{\text{SH}}^{(t+1)} = 1$ 和 $G_{a_i}^{(t+1)} = 0$ 。这个过程确保了 $\boldsymbol{G}$ 仍然在对应于源数据集中实体类型的 TR() 和 RE() 动作的列中反映原始的黄金注释，但同时结合了模型对其他动作的概率。然后，通过设置每个 $1 \leq t \leq T_{\text{final}}$ ，

$$L^{(t)} = -\frac{1}{|\mathcal{A}_{\widehat{\mathbb{E}}}|} \sum_{a_i \in \mathcal{A}_{\widehat{\mathbb{E}}}} \left(G_{a_i}^{(t)} \log\left(\sigma\left(u_{a_i}^{(t)}\right)\right) + \left(1 - G_{a_i}^{(t)}\right) \log\left(1 - \sigma\left(u_{a_i}^{(t)}\right)\right) \right)$$

样本的总损失计算为

$$L = \frac{1}{T_{\text{final}}} \sum_{t=1}^{T_{\text{final}}} L^{(t)} .$$

鉴于 $\boldsymbol{G}$ 的构建方式，这确保了上述损失函数的约束条件 i) 和 ii) 得以满足。

为了评估所提架构在命名实体识别任务中的性能，在基准数据集上进行了单任务实验，特别是 CoNLL-2003 英语子集和 GENIA 的实验。通过使用以前研究中广泛使用的六个生物医学数据集的集成训练模型来评估其多任务性能。为了证明 SRU-NER 在下游应用中的可行性，通过复制 Sänger et al., 2024 的实验设置，在交互语料库环境中评估模型。最后，进行了两个进一步的实验，以量化多任务模型对在测试语料库中未明确标注的实体类型的预测可靠性，从而更全面地评估它们的泛化能力。以下章节中使用的数据集以及相应的实验设置在附录 A 中有描述。训练细节可以在附录 B 中找到。为了评价目的，预测的提及被认为是真阳性，如果且仅当其跨度边界和实体类型与金标准注释完全匹配。结果使用提及级别的微 F1 分数为每个数据集报告。

3.4 单任务性能

两个单任务模型的结果如表 2 所示。所提出的模型在 CoNLL-2003 数据集上实现了 94.48 %

的微平均 F1 分数，在 GENIA 数据集上实现了 80.10 % 的微平均 F1 分数。这些结果非常接近当前的最佳水平（SOTA），展示了 SRU-NER 在平面和嵌套命名实体识别（NER）场景中的竞争力。然而，与我们的方法不同，被称为 SOTA 的模型在训练时使用了其各自数据集的训练和开发划分。这种训练数据的可用性差异可能导致观察到的性能差距，特别是在 GENIA 数据集中，额外的标注数据可能进一步有助于捕捉复杂的生物医学术语。

Dataset	SRU-NER	SOTA
CoNLL	94.48	94.6*, (Wang et al., 2021)
GENIA	80.10	81.53*, (Shen et al., 2023)

Table 2: 基准数据集中单任务模型的 Micro-F1 分数。数据集的实体数量可参见表 7 。* 符号表示该模型是在语料库的训练和开发集上训练的。

Dataset	SRU-NER	SOTA
BC2GM	85.43	85.48* (Sun et al., 2021)
BC4CHEMD	92.64	92.92* (Sun et al., 2021)
BC5CDR	90.61	91.90 (Zhang et al., 2023)
JNLPBA	77.12	78.93* (Sun et al., 2021)
Linnaeus	89.62	94.13 (Habibi et al., 2017)
NCBI Disease	89.25	90.04* (Sun et al., 2021)
Average	87.45	

Table 3: 在用于多任务模型（详见第 3.5 节）训练的数据集上训练的单任务模型的 Micro-F1 分数。SOTA 结果针对于单任务模型。星号符号表示模型是在更大的训练集上训练的。

3.5 多任务性能

在表 1 中，我们展示了在六个生物医学数据集的集合 $\{D_i\}_{i=1}^{6}$ 上训练的 SRU-NER 的结果，这些数据集注释了 $|\cup_i \mathbb{E}_i| = 8$ 种实体类型。由于有一些实体类型在多个数据集上被注释（例如，BC4CHEMD 和 BC5CDR 都注释了化学类型的提及），因此考虑了两种不同的评估方案，这两种方案在这些类型标签的解释上有所不同。回顾一下，模型根据来自集合的数据集 D_i 的测试分割中的句子，在不交集集 $\widehat{\mathbb{E}} = \sqcup_i \mathbb{E}_i$ 中推断出具有实体类型的提及情况，在以下情形中：

i) 独立评估，$\mathbb{E}_i \subset \widehat{\mathbb{E}}$ 类型的预测跨度与标准答案进行比较，而任何属于 $\widehat{\mathbb{E}} \setminus \mathbb{E}_i$ 类型的预测跨度都会被丢弃；

ii) 合并评估中，预测范围的实体类型被映射到 $\cup_i \mathbb{E}_i$ ，并且那些其映射类型不属于 $\mathbb{E}_i$ 的范围被丢弃；剩余的范围将与正确答案进行比较。

图 3 显示了模型对测试句子的预测示例，以及在两种评估场景中用于计算指标的文本跨度。

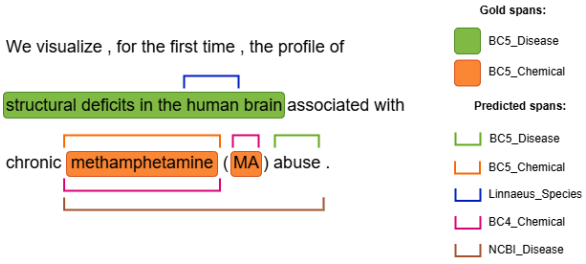

Figure 3: BC5CDR 语料库测试集中的句子示例 (Li et al., 2016) ，连同金标准跨度和由第 3.5 节描述的 MTL 模型预测的跨度一起注释。该模型在六个数据集上进行训练，涵盖八种实体类型 $\cup_i \mathbb{E}_i = \{\text{Chemical}, \text{Disease}, \ldots\}$ 。注意，这些类型中的某些类型在多个数据集中是共有的（即，“Chemical” 在 BC4CHEMD 和 BC5CDR 数据集中都有注释；“Disease” 在 BC5CDR 和 NCBI 数据集中都有注释）。SRU-NER 使用 11 种可能类型之一对跨度进行标记，通过将数据集名称连接到原始类型名称来构建，以致 $\widehat{\mathbb{E}} = \{\text{BC4\_Chemical}, \text{BC5\_Chemical}, \ldots\}$ 。在不相连的评估情况下，由于该句子来自 BC5CDR 语料库，指标的计算仅考虑其类型在 $\widehat{\mathbb{E}}$ 中以 BC5 缩写开头的跨度，结果是一个真阳性、一个假阳性和两个假阴性。在合并评估的情况下，类型在 $\widehat{\mathbb{E}}$ 中不以 “Chemical” 或 “Disease” 结尾的跨度被丢弃，剩余的跨度通过去掉数据集标识符将其类型映射为 $\cup_i \mathbb{E}_i$ 。在这些跨度中，该句子中有两个真阳性、两个假阳性和一个假阴性。

与先前的多任务学习（MTL）模型相比，所提出的模型在不相交的评估设置中取得了最佳或次佳的 F1 得分。这些结果是在不依赖于任务特定分类层 (Wang et al., 2018; Khan et al., 2020) 或训练多个单任务教师模型然后通过知识蒸馏到学生模型 (Moscato et al., 2023) 的情况下获得的。相反，单个统一模型直接从其各自的注释数据集中学习每个任务，同时保持其他任务的性能。这种方法实现了联合解码，从而消除了需要进行后处理步骤来解决跨度冲突的需求。

表格 3 展示了在多任务设置中用于每个数据集训练的单任务模型的 F1 分数，同时也包括 SOTA 参考。结果表明，所提出的模型在单任务设置中仍然具有竞争力。六个单任务 SRU-NER 模型的平均 F1 分数比多任务 SRU-NER 模型在不相交评估设置下的数据集平均 F1 高 0.82 个百分点。这与以往的研究结果一致，表明虽然多任务训练提高了模型在不同数据集上的鲁棒性，但与单任务模型相比，可能导致模型在特定语料中的表现下降 (Yin et al., 2024) 。为了进一步研究模型的泛化能力，下一节将在

跨语料设置中进行评估。

Dataset	Entity type	SRU-NER	Baseline
BioID	Species	62.41	58.21
MedMentions	Chemical	59.53	58.40
	Disease	62.48	62.18
tmVar3	Gene	90.38	87.87
Average		68.70	66.67

Table 4: 跨语料库实验的提及级别 F1 分数。SRU-NER 是在 8 个生物医学数据集的集成上进行训练，并在 3 个独立的语料库上进行评估。Baseline 指的是由 (Sänger et al., 2024) 获得的分数。最佳分数以粗体显示。

Training datasets	Chemical	Disease
Only BC5-Chemical	91.27	—
Only BC5-Disease	—	85.41
Both	91.81	86.10

Table 5: 在合成数据集上训练的模型在 BC5CDR 测试集上的全局预测 F1 分数。最佳分数用粗体显示。

3.6 跨语料库评估

表 4 显示了在交叉语料库评估中所提出模型的结果，重现了 Sänger et al., 2024 的实验设置。该模型在涵盖五种实体类型的九个数据集的集合上进行了训练，并在对其中四种类型注释的三个独立语料库上进行了评估。结果表明，SRU-NER 的表现平均比基线高出 2.03 %，其中物种 (4.2 %) 和基因 (2.51 %) 实体类型的改进尤为显著。这些发现突显了该模型的稳健性，并展示了其在下游应用中的潜力。供参考，在语料库内的 F1 分数提供在附录 C 中。

之前的实验评估了模型的局部预测能力。具体来说，当模型在一个集合 $\{D_i\}_{i=1}^{K}$ 上进行训练时，其中每个数据集 D_i 都针对实体类型 $\mathbb{E}_i$ 进行了标注，其性能是在一个测试数据集 D_{test} 上进行评估的，该数据集用一些 $j \in \{1,\ldots,K\}$ 的实体类型 $\mathbb{E}_{\text{test}} \subseteq \mathbb{E}_j$ 进行了标注。然而，模型会在 D_{test} 中生成对 $\cup_i \mathbb{E}_i$ 中所有实体类型的跨度的预测。为了评估其全局预测能力，有必要在一个用跨多个训练数据集实体类型的超集进行标注的数据集上测试模型。

首先，按照 Huang et al., 2019 的方法，从 BC5CDR 语料库中构建一个综合数据集。原始训练集被随机划分为两个不相交的子集：一个仅包含化学注释 (BC5-Chemical)，另一个仅包含疾病注释 (BC5-Disease)。关于这些综合数据集的更多细节见附录 A。两个单任务模型分别在每个子集上训练，而一个多任务模型则在两个子集上进行训练。所有模型都在 BC5CDR 语料库的原始测试集上进行评估。表 5 中的结果表明，在章节 **??** 中概述的训练策略有效地使模型能够在来自不同训练数据集的实体类型之间进行准确的全局预测。

其次，一个多任务模型在 CoNLL-2003 数据集和 BC5CDR 数据集上进行训练。这种方法生成的模型能够识别六种实体类型：四种来自通用领域 (LOC, MISC, ORG, PER)，两种来自生物医学领域 (Chemical, Disease)。为了评估模型在跨领域上的泛化能力，对其在 BC5CDR 数据集测试集上的通用领域实体类型预测以及在 CoNLL 数据集测试集上的生物医学实体类型预测进行了评估。多任务模型的结果见表格 6 的 SRU-NER-MTL 列。由于这些跨领域预测的黄金标注不可用，评估是由两位人工标注者手动进行的。在提供实体类型定义的情况下，他们独立评估模型的预测是否正确。对于两个单任务模型的预测也进行人类评估：一个在 CoNLL-2003 上训练并在 BC5CDR 测试集上评估 (SRU-NER-CoNLL)，另一个在 BC5CDR 上训练并在 CoNLL-2003 测试集上评估 (SRU-NER-BC5)。单任务和多任务模型的比较显示，多任务 SRU-NER 在识别域外跨度时，平均精确度提升了 25.4 %。例如，训练在生物医学实体类型上的单任务模型错误地将 lead 分类为 CoNLL-2003 句子中的化学物质：“Indonesian keeper Hendro Kartiko produced a string of fine saves to prevent the Koreans increasing their lead.” 而多任务模型没有犯这个错误。关于此实验的更多详细信息，见附录 D。

4 结论

这项工作提出了 SRU-NER，一种用于命名实体识别的新型架构，能够通过基于转换的解析方法处理嵌套实体。该模型整合了槽基递归单元 (SRU)，以维护过去操作的不断演变的表示，从而实现有效的实体提取。与传统的依赖于为不同实体类型设定单独模型的多任务学习方法不同，SRU-NER 采用统一的学习策略，允许单一模型从多个数据集中学习。此方法提高了对标注不一致性的适应性，并增强了跨领域的泛化能力。

实验结果表明，SRU-NER 在单任务和多任务环境中都表现出强大的性能，跨语料库评估和人工评估证实了其预测的稳健性。这些发现突显了为生物命名实体识别 (BioNER) 训练一个单一多任务模型的优势，并为未来研究指明了有前景的方向，包括嵌套实体识别的进步和领域适应性。

Entity	SRU-NER-CoNLL			SRU-NER-BC5			SRU-NER-MTL		
	P	R	F1	P	R	F1	P	R	F1
Chemical	24.71	87.76	38.57	—	—	—	75.00	9.18	16.36
Disease	25.25	83.33	38.76	—	—	—	88.46	38.33	53.49
LOC	—	—	—	98.25	88.89	93.33	100.00	96.83	98.39
ORG	—	—	—	80.00	80.00	80.00	86.36	71.25	78.08
PER	—	—	—	94.44	94.44	94.44	100.00	22.22	36.36

Table 6: 对三种模型在域外预测的人类评估。P 代表精确度，R 代表模拟的召回率，F1 代表用前两项指标计算的 F1 分数。关于如何计算这些指标的详细信息可以在附录 D 中找到。

5

局限性

虽然所提出的 SRU-NER 架构在一般和生物医学领域的命名实体识别中显示了有效性，但其在其他领域（如法律或金融）中的表现尚未评估。此外，由于评估在社区可用的生物医学数据集上进行，这些发现的普遍性可能有限，因为这可能未能充分反映现实世界生物医学文本的多样性。最后，跨领域场景下对整体预测能力的评估依赖于人工注释，给评价引入了一定程度的主观性。虽然模型取得了有竞争力的结果，但我们注意到并未进行广泛的超参数搜索。更系统地调整超参数可能带来进一步的改进。此外，训练策略提供了精细化的机会，特别是在多任务学习框架中使用的采样策略方面。

这项研究得到了葡萄牙复苏与韧性计划资助，通过项目 C645008882-00000055（即，负责任人工智能中心）。

References

Cecilia Arighi, Lynette Hirschman, Thomas Lemberger, Samuel Bayer, Robin Liechti, Donald Comeau, and Cathy Wu. 2017. Bio-id track overview. In *BioCreative VI Challenge Evaluation Workshop*, volume 482, page 376.

Pratyay Banerjee, Kuntal Kumar Pal, Murthy Devarakonda, and Chitta Baral. 2021. Biomedical Named Entity Recognition via Knowledge Guidance and Question Answering. *ACM Trans. Comput. Healthcare*, 2(4):33:1–33:24.

Nigel Collier, Tomoko Ohta, Yoshimasa Tsuruoka, Yuka Tateisi, and Jin-Dong Kim. 2004. Introduction to the bio-entity recognition task at JNLPBA. In *Proceedings of the International Joint Workshop on Natural Language Processing in Biomedicine and its Applications (NLPBA/BioNLP)*, pages 73–78, Geneva, Switzerland. COLING.

Alexis Conneau, Kartikay Khandelwal, Naman Goyal, Vishrav Chaudhary, Guillaume Wenzek, Francisco Guzmán, Edouard Grave, Myle Ott, Luke Zettlemoyer, and Veselin Stoyanov. 2020. Unsupervised cross-lingual representation learning at scale. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 8440–8451, Online. Association for Computational Linguistics.

Gamal Crichton, Sampo Pyysalo, Billy Chiu, and Anna Korhonen. 2017. A neural network multi-task learning approach to biomedical named entity recognition. *BMC Bioinformatics*, 18(1):368.

Rezarta Islamaj Doğan, Robert Leaman, and Zhiyong Lu. 2014. NCBI disease corpus: a resource for disease name recognition and concept normalization. *J Biomed Inform*, 47:1–10.

Chris Dyer, Miguel Ballesteros, Wang Ling, Austin Matthews, and Noah A. Smith. 2015. Transition-based dependency parsing with stack long short-term memory. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 334–343, Beijing, China. Association for Computational Linguistics.

Octavian-Eugen Ganea and Thomas Hofmann. 2017. Deep joint entity disambiguation with local neural attention. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 2619–2629, Copenhagen, Denmark. Association for Computational Linguistics.

Martin Gerner, Goran Nenadic, and Casey M. Bergman. 2010. Linnaeus: A species name identification system for biomedical literature. *BMC Bioinformatics*, 11(1):85.

Nathan Greenberg, Trapit Bansal, Patrick Verga, and Andrew McCallum. 2018. Marginal likelihood training of BiLSTM-CRF for biomedical named entity recognition from disjoint label sets. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 2824–2829, Brussels, Belgium. Association for Computational Linguistics.

Harsha Gurulingappa, Roman Klinger, Martin Hofmann-Apitius, and Juliane Fluck. 2010. An empirical evaluation of resources for the identification of diseases and adverse effects in biomedical literature. In *2nd Workshop on Building and evaluating resources for biomedical text mining (7th*

edition of the Language Resources and Evaluation Conference), Valetta, Malta.

Maryam Habibi, Leon Weber, Mariana Neves, David Luis Wiegandt, and Ulf Leser. 2017. Deep learning with word embeddings improves biomedical named entity recognition. *Bioinformatics*, 33(14):i37–i48.

Xiao Huang, Li Dong, Elizabeth Boschee, and Nanyun Peng. 2019. Learning a unified named entity tagger from multiple partially annotated corpora for efficient adaptation. In *Proceedings of the 23rd Conference on Computational Natural Language Learning (CoNLL)*, pages 515–527, Hong Kong, China. Association for Computational Linguistics.

Rezarta Islamaj, Robert Leaman, Sun Kim, Dongseop Kwon, Chih-Hsuan Wei, Donald C Comeau, Yifan Peng, David Cissel, Cathleen Coss, Carol Fisher, Rob Guzman, Preeti Gokal Kochar, Stella Koppel, Dorothy Trinh, Keiko Sekiya, Janice Ward, Deborah Whitman, Susan Schmidt, and Zhiyong Lu. 2021a. NLM-Chem, a new resource for chemical entity recognition in PubMed full text literature. *Sci Data*, 8(1):91.

Rezarta Islamaj, Chih-Hsuan Wei, David Cissel, Nicholas Miliaras, Olga Printseva, Oleg Rodionov, Keiko Sekiya, Janice Ward, and Zhiyong Lu. 2021b. NLM-Gene, a richly annotated gold standard dataset for gene entities that addresses ambiguity and multi-species gene recognition. *J Biomed Inform*, 118:103779.

Muhammad Raza Khan, Morteza Ziyadi, and Mohamed AbdelHady. 2020. Mt-bioner: Multi-task learning for biomedical named entity recognition using deep bidirectional transformers. *Preprint*, arXiv:2001.08904.

J.-D. Kim, T. Ohta, Y. Tateisi, and J. Tsujii. 2003. Genia corpus—a semantically annotated corpus for bio-textmining. *Bioinformatics*, 19:i180–i182.

Corinna Kolarik, Roman Klinger, Christoph M. Friedrich, Martin Hofmann-Apitius, and Juliane Fluck. 2008. Chemical Names: Terminological Resources and Corpora Annotation. In *Workshop on Building and evaluating resources for biomedical text mining (6th edition of the Language Resources and Evaluation Conference)*, pages 51–58, Marrakech, Morocco.

Martin Krallinger, Obdulia Rabal, Florian Leitner, Miguel Vazquez, David Salgado, Zhiyong Lu, Robert Leaman, Yanan Lu, Donghong Ji, Daniel M. Lowe, Roger A. Sayle, Riza Theresa Batista-Navarro, Rafal Rak, Torsten Huber, Tim Rocktäschel, Sérgio Matos, David Campos, Buzhou Tang, Hua Xu, Tsendsuren Munkhdalai, Keun Ho Ryu, S. V. Ramanan, Senthil Nathan, Slavko Žitnik, Marko Bajec, Lutz Weber, Matthias Irmer, Saber A. Akhondi, Jan A. Kors, Shuo Xu, Xin An, Utpal Kumar Sikdar, Asif Ekbal, Masaharu Yoshioka, Thaer M. Dieb, Miji Choi, Karin Verspoor, Madian Khabsa, C. Lee Giles, Hongfang Liu, Komandur Elayavilli Ravikumar, Andre Lamurias, Francisco M. Couto, Hong-Jie Dai, Richard Tzong-Han Tsai, Caglar Ata, Tolga Can, Anabel Usié, Rui Alves, Isabel Segura-Bedmar, Paloma Martínez, Julen Oyarzabal, and Alfonso Valencia. 2015. The chemdner corpus of chemicals and drugs and its annotation principles. *Journal of Cheminformatics*, 7(1):S2.

Jiao Li, Yueping Sun, Robin J Johnson, Daniela Sciaky, Chih-Hsuan Wei, Robert Leaman, Allan Peter Davis, Carolyn J Mattingly, Thomas C Wiegers, and Zhiyong Lu. 2016. BioCreative V CDR task corpus: a resource for chemical disease relation extraction. *Database (Oxford)*, 2016.

Jing Li, Aixin Sun, Jianglei Han, and Chenliang Li. 2022. A survey on deep learning for named entity recognition. *IEEE Transactions on Knowledge and Data Engineering*, 34(1):50–70.

Ling Luo, Po-Ting Lai, Chih-Hsuan Wei, Cecilia N Arighi, and Zhiyong Lu. 2022. Biored: a rich biomedical relation extraction dataset. *Briefings in Bioinformatics*, 23(5):bbac282.

Ling Luo, Chih-Hsuan Wei, Po-Ting Lai, Robert Leaman, Qingyu Chen, and Zhiyong Lu. 2023. AIONER: all-in-one scheme-based biomedical named entity recognition using deep learning. *Bioinformatics*, 39(5):btad310.

Zita Marinho, Afonso Mendes, Sebastião Miranda, and David Nogueira. 2019. Hierarchical nested named entity recognition. In *Proceedings of the 2nd Clinical Natural Language Processing Workshop*, pages 28–34, Minneapolis, Minnesota, USA. Association for Computational Linguistics.

Tahir Mehmood, Alfonso Gerevini, Alberto Lavelli, and Ivan Serina. 2019. Leveraging multi-task learning for biomedical named entity recognition. In *AI*IA 2019 – Advances in Artificial Intelligence*, pages 431–444, Cham. Springer International Publishing.

Sunil Mohan and Donghui Li. 2019. Medmentions: A large biomedical corpus annotated with umls concepts. *Preprint*, arXiv:1902.09476.

Vincenzo Moscato, Marco Postiglione, Carlo Sansone, and Giancarlo Sperlí. 2023. Taughtnet: Learning multi-task biomedical named entity recognition from single-task teachers. *IEEE Journal of Biomedical and Health Informatics*, 27(5):2512–2523.

Evangelos Pafilis, Sune P Frankild, Lucia Fanini, Sarah Faulwetter, Christina Pavloudi, Aikaterini Vasileiadou, Christos Arvanitidis, and Lars Juhl Jensen. 2013. The SPECIES and ORGANISMS resources for fast and accurate identification of taxonomic names in text. *PLoS One*, 8(6):e65390.

Yesol Park, Gyujin Son, and Mina Rho. 2024. Biomedical flat and nested named entity recognition: Methods, challenges, and advances. *Applied Sciences*, 14(20).

Abhishek Sharma, Amrita, Sudeshna Chakraborty, and Shivam Kumar. 2022. Named entity recognition in natural language processing: A systematic review. In *Proceedings of Second Doctoral Symposium on Computational Intelligence*, pages 817–828, Singapore. Springer Singapore.

Yongliang Shen, Kaitao Song, Xu Tan, Dongsheng Li, Weiming Lu, and Yueting Zhuang. 2023. DiffusionNER: Boundary diffusion for named entity recognition. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 3875–3890, Toronto, Canada. Association for Computational Linguistics.

Larry Smith, Lorraine K. Tanabe, Rie Johnson nee Ando, Cheng-Ju Kuo, I-Fang Chung, Chun-Nan Hsu, Yu-Shi Lin, Roman Klinger, Christoph M. Friedrich, Kuzman Ganchev, Manabu Torii, Hongfang Liu, Barry Haddow, Craig A. Struble, Richard J. Povinelli, Andreas Vlachos, William A. Baumgartner, Lawrence Hunter, Bob Carpenter, Richard Tzong-Han Tsai, Hong-Jie Dai, Feng Liu, Yifei Chen, Chengjie Sun, Sophia Katrenko, Pieter Adriaans, Christian Blaschke, Rafael Torres, Mariana Neves, Preslav Nakov, Anna Divoli, Manuel Maña-López, Jacinto Mata, and W. John Wilbur. 2008. Overview of biocreative ii gene mention recognition. *Genome Biology*, 9(2):S2.

Cong Sun, Zhihao Yang, Lei Wang, Yin Zhang, Hongfei Lin, and Jian Wang. 2021. Biomedical named entity recognition using bert in the machine reading comprehension framework. *Journal of Biomedical Informatics*, 118:103799.

Mario Sänger, Samuele Garda, Xing David Wang, Leon Weber-Genzel, Pia Droop, Benedikt Fuchs, Alan Akbik, and Ulf Leser. 2024. Hunflair2 in a cross-corpus evaluation of biomedical named entity recognition and normalization tools. *Bioinformatics*, 40(10):btae564.

Erik F. Tjong Kim Sang and Fien De Meulder. 2003. Introduction to the CoNLL-2003 shared task: Language-independent named entity recognition. In *Proceedings of the Seventh Conference on Natural Language Learning at HLT-NAACL 2003*, pages 142–147.

Xinyu Wang, Yong Jiang, Nguyen Bach, Tao Wang, Zhongqiang Huang, Fei Huang, and Kewei Tu. 2021. Automated concatenation of embeddings for structured prediction. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 2643–2660, Online. Association for Computational Linguistics.

Xuan Wang, Yu Zhang, Xiang Ren, Yuhao Zhang, Marinka Zitnik, Jingbo Shang, Curtis Langlotz, and Jiawei Han. 2018. Cross-type biomedical named entity recognition with deep multi-task learning. *Bioinformatics*, 35(10):1745–1752.

Chih-Hsuan Wei, Alexis Allot, Kevin Riehle, Aleksandar Milosavljevic, and Zhiyong Lu. 2022. tmvar 3.0: an improved variant concept recognition and normalization tool. *Bioinformatics*, 38(18):4449–4451.

Chih-Hsuan Wei, Hung-Yu Kao, and Zhiyong Lu. 2015. GNormPlus: An integrative approach for tagging genes, gene families, and protein domains. *Biomed Res Int*, 2015:918710.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Remi Louf, Morgan Funtowicz, Joe Davison, Sam Shleifer, Patrick von Platen, Clara Ma, Yacine Jernite, Julien Plu, Canwen Xu, Teven Le Scao, Sylvain Gugger, Mariama Drame, Quentin Lhoest, and Alexander Rush. 2020. Transformers: State-of-the-art natural language processing. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 38–45, Online. Association for Computational Linguistics.

Hang Yan, Yu Sun, Xiaonan Li, and Xipeng Qiu. 2023. An embarrassingly easy but strong baseline for nested named entity recognition. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 1442–1452, Toronto, Canada. Association for Computational Linguistics.

Michihiro Yasunaga, Jure Leskovec, and Percy Liang. 2022. LinkBERT: Pretraining language models with document links. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 8003–8016, Dublin, Ireland. Association for Computational Linguistics.

Yu Yin, Hyunjae Kim, Xiao Xiao, Chih Hsuan Wei, Jaewoo Kang, Zhiyong Lu, Hua Xu, Meng Fang, and Qingyu Chen. 2024. Augmenting biomedical named entity recognition with general-domain resources. *Journal of Biomedical Informatics*, 159:104731.

Sheng Zhang, Hao Cheng, Jianfeng Gao, and Hoifung Poon. 2023. Optimizing bi-encoder for named entity recognition via contrastive learning. In *The Eleventh International Conference on Learning Representations*.

A 数据集和实验设置

对于 CoNLL-2003 (Tjong Kim Sang and De Meulder, 2003) 的英文子集，使用了原始数据集的划分，这些划分以事先标记的格式提供。对于 GENIA 数据集，采用了 Yan et al., 2023 的划分。这些数据集的每个划分的实体数量可以在表格 7 中找到。

Dataset	Entity Type	Train	Dev	Test
CONLL	LOC	7,140	1,837	1,668
	MISC	3,438	922	702
	ORG	6,321	1,341	1,661
	PER	6,600	1,842	1,617
GENIA	Cell Line	3,069	372	403
	Cell Type	5,854	576	578
	DNA	7,707	1,161	1,132
	Gene or protein	28,874	2,466	2,900
	RNA	699	139	106

Table 7: 第 3.4 节单任务实验中使用的数据集的统计信息。

为了训练一个多任务模型，使用了六个生物医学数据集：BC2GM (Smith et al., 2008) 、BC4CHEMD (Krallinger et al., 2015) 、BC5CDR (Li et al., 2016) 、JNLPBA (Collier et al., 2004) 、Linnaeus (Gerner et al., 2010) 和 NCBI Disease (Doğan et al., 2014) 。数据集的划分（表格 8 ）遵循 Crichton et al., 2017 所建立的划分，这些划分在先前的研究（包括 Wang et al., 2018; Huang et al., 2019; Khan et al., 2020; Moscato et al., 2023 ）中已被广泛使用。

Dataset	Entity Type	Train	Dev	Test
BC2GM	Gene or protein	15,035	3,032	6,243
BC4CHEMD	Chemical	29,263	29,305	25,210
BC5CDR	Chemical	5,114	5,239	5,277
	Disease	4,169	4,224	4,394
JNLPBA	Cell Line	3,369	389	490
	Cell Type	6,162	522	1,906
	DNA	8,416	1,040	1,045
	Gene or protein	27,015	2,379	4,988
	RNA	844	106	118
Linnaeus	Species	2,079	700	1,412
NCBI Disease	Disease	5,111	779	952

Table 8: 用于第 3.5 节多任务实验的数据集的统计数据。

在前述实验中，模型在各自的训练拆分上进行训练，检查点选择是在开发拆分上进行的，评估是在测试拆分上进行的。

对于跨语料库评估，复制了 Sänger et al., 2024 的实验设置。使用九个数据集 [8] 的集合训练一个多任务模型：BioRED (Luo et al., 2022) 、GNormPlus (Wei et al., 2015) 、Linnaeus (Gerner et al., 2010) 、NCBI Disease (Doğan et al., 2014) 、NLM-Chem (Islamaj et al., 2021a) 、NLM-Gene (Islamaj et al., 2021b) 、S800 (Pafilis et al., 2013) 、SCAI Chemical (Kolarik et al., 2008) 和 SCAI Disease (Gurulingappa et al., 2010) 。模型在训练集上进行训练，并在开发集上选择检查点。评估是在一个独立语料库上进行的，该语料库由三个数据集的完整注释数据组成：BioID (Arighi et al., 2017) 、MedMentions (Mohan and Li, 2019) 和 tmVar3 (Wei et al., 2022) 。训练语料库和独立测试语料库的数据集统计信息分别可以在表 9 和表 10 中找到。

Dataset	Entity Type	Train	Dev	Test
BioRED	Cell Line	103	22	50
	Chemical	2,830	818	751
	Disease	3,643	982	917
	Gene	4,404	1,087	1,170
	Species	1,429	370	393
GNormPlus	Gene	4,964	504	4,468
Linneaus	Species	1,725	206	793
NCBI Disease	Disease	4,083	666	2,109
NLM-Chem	Chemical	21,102	5,223	11,571
NLM-Gene	Gene	11,209	1,314	2,687
S800	Species	2,236	410	1,079
SCAI Chemical	Chemical	852	83	375
SCAI Disease	Disease	1,281	250	710

Table 9: 第 3.6 节中跨语料库评估场景中使用的训练语料库的统计数据。

Dataset	Entity Type	Number of mentions
BioID	Species	7,939
tmVar3	Gene	4,059
MedMentions	Disease	19,298
	Chemical	19,198

Table 10: 用于第 3.6 节中描述的跨语料库评估的语料库统计数据。

最终，为了评估模型的全局预测能力，按照 (Huang et al., 2019) 实验设置，从 BC5CDR 语料库中提取了合成数据集。原始训练集被随机划分成两个不相交的子集：BC5-疾病（仅包含疾病注释）和 BC5-化学（仅包含化学注释）。开发集的分割也遵循了相同的程序。这些合成数据集的统计数据见表 11 。通过在 BC5-疾病和 BC5-化学子集上训练模型并在 BC5CDR 语料库的完整测试集上评估它们，我们可以测试模型的全局预测能力，如 **??** 部分所述。

[8] 数据集是在 2025 年 2 月从 `https://github.com/flairnlp/flair` 获取的。它们的划分和预处理选择被复现了。

Dataset	Entity Type	Train	Dev
BC5-Disease	Disease	2,172	2,279
BC5-Chemical	Chemical	2,459	2,665

Table 11: 用于评估全局预测能力的合成数据集统计。

B 训练细节

Hyperparameter	GENIA	Others
# epochs	100	100
Early stop	30	30
Batch size	16	16
Max. # tokens	405	405
Gradient norm clipping	1.0	1.0
Dropout on logits	0.1	0.1
SRU module		
# latent embeddings (multiplier)	10	2
Half-context for pos. embeddings	240	150
Dropout on pos. embeddings	0.2	0.2
Dropout on latent embeddings	0.2	0.2
Encoder optimizer		
LR	3e-5	2e-5
Weight decay	1e-3	1e-3
Warm up (in epochs)	1	1
Actions generation cycle optimizer		
LR	3e-4	3e-4
Weight decay	1e-3	1e-3
Warm up (in epochs)	0.5	0.5

Table 12: 实验中使用的超参数。“其他”列指的是除了在 GENIA 数据集上的单任务之外的每个实验。

所有模型均使用 PyTorch 张量库开发，并在单个 NVIDIA A100 80GB GPU 上进行训练。编码器模块和动作生成模块使用两个独立的 AdamW 优化器进行调整，并进行线性预热，设置了不同的初始学习率和权重衰减。两个优化器都设置为 $\beta_1 = 0.9$, $\beta_2 = 0.98$ 和 $\epsilon = 10^{-6}$ 。模型通过开发集上的性能基于早期停止进行训练。[9] 所有实验的超参数可以在表 12 中找到。此外，尽管在 GENIA 数据集上的单任务实验中方程式 (5) 中部分 3.3 的令牌缩放参数 α 是经过训练的，但在所有其他实验中它都被冻结并设置为 1 。

编码器模块是在 HuggingFace transformers 库的基础上构建的 (Wolf et al., 2020) 。具体来说，所有使用生物医学语料库训练的模型都使用了来自 Yasunaga et al., 2022 的 `LinkBERT-large` 编码器，而在 CoNLL-2003 数据集上训练的单任务模型则使用了由 Conneau et al., 2020 引入的 `xlm-roberta-large` 编码器。

[9]在多任务模型的情况下，多个数据集被标记为相同的实体类型（如第 3.5 节和第 3.6 节中的模型），尽管为了训练目的将实体类型视为不相交，但在第 3.5 节开头所述，验证集中用于检查点选择的 F1 分数是通过合并这些类型来计算的。

C 在用于跨语料库实验的数据集上的单任务表现

Dataset	Merged	Disjoint
BioRED	90.73	90.90
GNormPlus	85.00	86.00
Linnaeus	78.16	92.23
NCBI Disease	85.69	85.70
NLM-Chem	84.42	85.65
NLM-Gene	88.35	88.13
S800	74.24	75.79
SCAI Chemical	85.21	85.64
SCAI Disease	80.78	82.14

Table 13: 用于第 3.6 部分跨语料库评估实验的模型的语料库内微-F1 分数。

D 跨领域环境中对全局预测的人类评估

为了评估模型跨领域泛化的能力，训练了三个模型：

- SRU-NER-CoNLL：在 CoNLL 语料库上训练的单任务模型；
- SRU-NER-BC5：一个在 BC5CDR 语料库上训练的单任务模型；
- SRU-NER-MTL：一个在两个语料库上训练的多任务模型。

所有模型都使用来自 Yasunaga et al., 2022 的 `LinkBERT-large` 编码器进行了训练。为了评估跨领域的泛化能力，能够识别通用领域实体类型的模型（SRU-NER-CoNLL 和 SRU-NER-MTL）被用于标注生物医学语料库的测试分割，而在生物医学实体类型上训练的模型（SRU-NER-BC5 和 SRU-NER-MTL）则用于标注通用领域语料库的测试分割。由于这些域外预测的黄金标注不可用，两位语言学家手动评估了它们的正确性。每种实体类型的标注者间一致性在表 14 中报告。

Entity	Agreement (%)
Chemical	92.98
Disease	91.09
LOC	100.00
ORG	87.76
PER	88.89

Table 14: 针对被评估实体类型的标注者间一致性。

基于两位人工标注者对预测跨度的准确评估，通过将正确识别的跨度与预测跨度总数的

比值计算出每个模型、实体类型和语言学家的精确度得分。还通过考虑在三种模型的所有预测中至少被一位标注者认为正确的每种实体类型的跨度总数，计算出每个模型、实体类型和语言学家的模拟召回率。最终，通过对两位人工标注者的数据取平均值，获得每个模型和实体类型的精确度和模拟召回率。

结果可以在主要文本的表格 6 中找到。可以看到，多任务模型的精确度得分在所有实体类型中都高于单任务模型，而多任务模型的召回率值在除 ORG 以外的所有实体类型中都较差。

为了便于参考，三种模型在语料库中的表现如表 15 所示。

Model	CoNLL	BC5CDR
SRU-NER-CoNLL	90.51	—
SRU-NER-BC5	—	90.61
SRU-NER-MT	91.01	90.51

Table 15: 在跨域环境下，用于评估全球预测的三个模型的语料库内表现。单任务模型 SRU-NER-BC5 与第 3.5 节多任务实验中用于比较的模型相同。