为上下文显式独立建模的中文实体识别方法[①]

陈点[②]* 曹逸轩* 罗平[③]*[△]

(* 中国科学院智能信息处理重点实验室 中科院计算所 北京 100190)

(* 中国科学院大学 北京 100190)

([△] 鹏城实验室 深圳 518066)

摘 要 现有中文命名实体识别模型在公开数据集上的表现相对成熟，但现有研究指出，模型过度依赖实体文本的字面特征，而上下文对实体识别的影响却未得到重视。现有模型在简单的泛化测试中表现较差，为此本文提出为上下文显式独立建模，令模型对上下文与实体字面信息进行区分，进一步提出了相应的数据增强方法以训练模型中的上下文模块、实体字面模块和综合模块。实验结果表明，本文提出的方法在不损失测试集识别效果的情况下，明显改善了模型在不变性测试中的表现，较基准模型降低了 2.3% 的失败率。

关键词 自然语言处理，中文命名实体识别，上下文独立建模，数据增强

0 引言

命名实体识别 (Named Entity Recognition, NER) 作为自然语言处理领域的基石任务，已经具有相对成熟的技术路线。现有方法[1]在各实体数据集上都有不错的表现，然而近期研究[2]指出，现有模型在判断实体时会过于依赖文本的字面特征，这导致模型在一些简单的语言理解能力测试中反而表现不佳。例如，将一个句子中的某个人名改为其它人名时，现有命名实体模型却时常无法保持与原先一致的预测。现有工作提出[3]模型训练时会将实体的标签信息记忆在字的表征中，会导致模型面对未在训练样本中出现过的文本时，显著降低预测精度[4]。关于这种现象的讨论产生了面向模型泛化能力的新评测方式 CheckList[5]。

本文提出，模型在预测实体时应当判断需要依赖其字面 (surface form, name) 还是借助上下文 (context features, context) 的特征信息，进而提出了一种自适应地平衡字面与上下文信息的命名实体识别模型 NCModel (Name-and-Context Model)。该模型在字面表征的基础上，引入了独立建模上下文的编码模块，以引入自注意力机制的填空 (Cloze) 编码作为上下文的表征，并设置了可训练的平衡参数用于表征融合。

该方案的主要难点为，现有的命名实体数据集不具备用于指出判断每个实体时“应依据字面还是上下文”的标签。为此，本文设计了相应的数据增强方法生成伪标签用于训练。基于公开数据集料构建的不变性测试中，实验结果印证了提出的方法能够为模型带来泛化能力的提升，较之基准模型失败率降低了 2.3%。

[①] 国家重点研发计划 (2022YFB2702502)，国家自然科学基金项目 (62076231, 62206265)，国家博士后基金 (2021M703271) 资助项目。

[②] 男，1994 年生，博士生；研究方向：自然语言处理，文本信息抽取，中文文本校对等；E-mail: chendian@ict.ac.cn

[③] 通信作者，E-mail: luop@ict.ac.cn

（收稿日期：2022-12-21）

1　研究背景

本节将对本文中涉及的背景知识与相关研究进行介绍，进而根据现有命名实体识别模型的现状分析归纳，从文本表征编码的角度入手提出新的网络构建方式。

1.1　实体识别在不变性测试中的困境

表 1　不变性测试样例

目标文本	人类判断	模型判断
李先生 是这里最好的医生之一。	人名 √	人名 √
我今天晚饭打算去 必胜客 吃。	餐厅名 √	餐厅名 √
我今天晚饭打算去 李先生 吃。	餐厅名 √ (根据上下文判断)	人名 × (字面难以判断)

人类理解文本时，会借助上下文推断出未曾见过的字面属于何种类别，近期研究[2]指出现有 NER 模型在这一点上仍有不足。Ribeiro 等人[5]提出，固定测试集上的传统准确率指标高估了模型的性能，于是提出了 CheckList 评估列表。其中，不变性测试 (Invariance Test, InvTest) 期望模型能够模仿这种理解方式，在样本中出现标签保留扰动 (Label-preserving Perturbations) 时，仍能保持与扰动前相同预测的结果。测试发现，常见的基于条件随机场 (Conditional Random Field, CRF) 的方法[6]在此类测试中得出了高达 19.0% 和 8.1% 的错误率，广泛使用的文本预训练模型[7] (Pre-trained Textual Language Models, PTM)在不变性测试中也都有超过 10% 的失败率，这也许揭示着现有的 NER 模型不具备此类理解能力。以此作参考，本文为 NER 模型应用这一测试。表 1 最后一行带下划线文本为提及 (mention) 的字面“李先生”，此处指牛肉面餐厅品牌，应根据上下文“在……吃饭”判断为“餐厅名”，但现有模型会将其错误分类为“人名”。

判断实体类别时不能仅凭字面文本，根据上下文的不同，一个实体提及可能对应着人名或地名等实体。Field 和 Tsvetkov[3]认为，文本模型会学习其字面中类似于“先生”的标志性文字进行“捷径学习” (Short-cut Learning)。对于命名实体识别任务，Fu 等人[8]分析指出，现有模型预测实体时，假如提及的字面文本在训练集中出现过，预测准确度会明显提高。反之，当一段字面文本从未在训练集中出现过，模型表现会明显下降。这体现出已有 NER 模型也许并未掌握更加具有泛化能力的学习途径。本文致力于解决这一挑战：即尝试构建一个能够正确地注重上下文信息的模型，以提升现有中文实体识别模型的不变性能力。

1.2　命名实体识别的研究现状

命名实体识别作为自然语言处理的基石任务，广泛应用于各类下游任务[9]中。现有的中文命名实体识别模型相对成熟[6]，在各类实体数据集上都能带来令人满意的表现。近年来，中文 NER 模型更多地考虑如何引入外部知识，为编码的字符赋予更加丰富的语义信息。截至 2022 年初，常见数据集上的最优模型中，DCSAN[1] 和现有晶格 (lattice-based[10][11]) 建模方法借助外部词典引入构词知识，为模型提供成词信息并引入词语义；FGN[12] 则借助字型 (glyph) 特征，基于中文构词表意的特点，将图形含义引入字表征。

将不同角度的外部知识融入字句表征，是一个长期探索与尝试的过程，但因多种来源的特征堆积导致模型笨重、引入特征的做法种类繁杂等问题，这些方法暂难以广泛普及到实际应用中。目前尚未出现稳定的外部特征引入方案，主流做法为借助 PTM[7] 或字、词粒度的长短期记忆神经网络 (Long Short-term Memory, LSTM[6]) 等用于序列标注 (Sequence Labeling)。本文暂不关注如何为表征中融入更多知识或信息，而是着眼于网络设计的角度构建通用的方法，用于现有模型架构学习何时应借助上下文分析并作出判断。

前沿的中文 NER 模型 Boundary-Smoothing[13] 和 W^{2}NER[14] 转而考虑重新定义命名实体标注数据集的标签。这些方法不为每个字符位置设置实体标签，而是根据实体的起止位置训练模型预测自定义的弱正例标签[13]和字符间逻辑关系标签[14]，以学习实体边界从而提升模型准确度。这类模型面向分类方法与网络设计，不与“追加语料及外部知识”的方法相冲突，可以期待未来两类方案的融合效果。

在英文命名实体识别任务中，基于跨度 (span-based[15][16]) 的 NER 模型逐渐登上舞台，这类方法为文本中的每个跨度逐一分类，便于获取上下文表征，故本文参考了这类方法。然而较之序列标注，需要模型获取表征的跨度数量有平方级的增长，本文 2.2 节提出了基于跨度打分的筛选方法。

2 上下文显式建模的模型

本文期望引入实体上下文的独立建模，以缓解当前 NER 模型过于依赖实体字面的问题，改善模型在不变性测试中的表现。前文提到，无论基于双向 LSTM[6] 或 PTM[7] 编码，字符表征中不可避免地包含了全句所有字的信息。为了追加不受字面影响的额外判断途径，本文提出了“不感知字面文本”(name-unaware) 的上下文表征独立编码模块 (the explicitly-modeling context module, 后称上下文模块)，与广义的上下文相比，编码表征时不会感知字面的文本信息。对于每个跨度，模型将综合考量字面与上下文的表征进行标签分类。

已有方法通过拼接提及周边的字表征[17]、或是匹配上下文中出现的标志词[18]用于上下文表征，它们或是依赖的字嵌入 (character-level embedding) 表征能力不够，或是需要依赖人工标注标志词才能生效。由于上下文表征较之字面表征而言，对于提及的特征归纳能力较差，现有的上下文表征分类方法[4]效果依然远不如主要基于字面文本分类的模型。本文尝试让字面与上下文的表征各司其职，在信息层面互相补足。如何不感知字面地获取上下文表征，并令其与现有网络提供的字面表征更好地协同工作，成为了模型网络设计中的重要挑战。基于上述问题，本文为标签分类确立了如下步骤：对于每个跨度，编码获取字面表征；借助上下文模块获取不感知字面的上下文表征；基于两种表征，模型判断当前跨度应以何种比例 (即平衡参数) 依赖上下文与字面信息；最终，借助融合后的表征作跨度分类。

2.1 模型总览

给定目标实体标签的集合 $\mathcal{E}$，NER 模型从目标文本 $x = (w_1, w_2, \dots, w_n)$ 中提取出所有所需类型的实体，其结果是一个实体集合 $\hat{E} = \{(i, j, t) \mid 0 \le i \le j < |x|\}, t \in \mathcal{E}$。序列标注方法[1][6]会同时给出所有字符的预测结果，在此之上构建实体输出。本文提出了上下文表征独立建模，由于需要获得每个提及跨度的上

下文表征，选择了基于跨度的方法[16]。从句子中获取的跨度 $(w_i, \dots, w_j)$ 可表示为 $s = (i, j)$。将跨度集合 S 中每个跨度逐一以标签集 $L = \mathcal{E} \cup \{\text{None}\}$ 分类，其中 None 指跨度被预测为“非实体”标签。

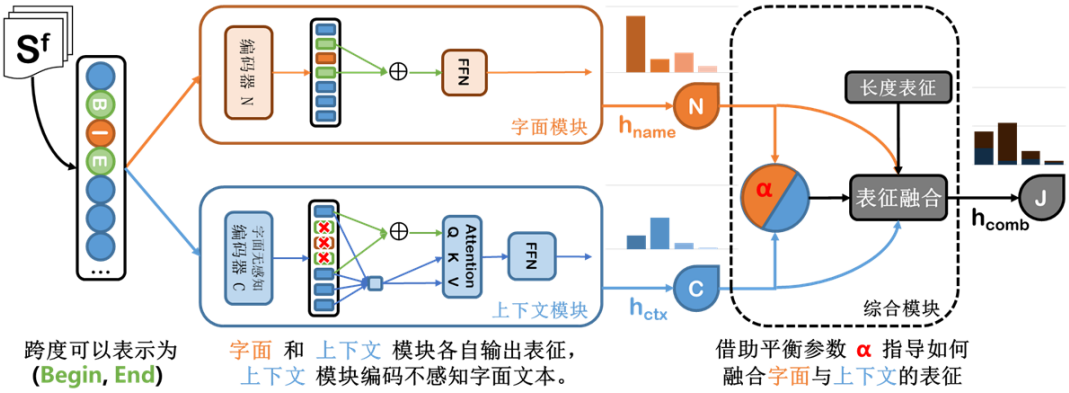

图 1 NCModel 模型网络结构

基于上述设计思路，本文实现了称作 NCModel (Name-and-Context Model) 的模型，意指模型分类时能够同时考虑字面与上下文的信息，并掌握判断目标跨度时更应该依赖其中哪类信息的能力。本节将根据实体识别模型的执行流程，依次介绍跨度候选 S^f 的获取以及三个不同目标的分类模块。

2.2 跨度候选

模型通过枚举从文本中获得跨度集合 S，跨度选择步骤通过筛选获得跨度的候选子集用于训练或预测，即图 1 左上的候选跨度集合 $S^f \subset S$。现有方法[16]借助长度限制，留存长度不超过特定值 l_{sp} 的跨度集 $S^f = \{s \mid s.length() \leq l_{sp}, s \in S\}$。这在英语语料中十分有效，设置为 6 或 10 个词长即可[16]。但跨度长度是在词或子词 (subword) 的层面上进行衡量，中文语料中跨度较长，往往因数量过多而导致训练与推理过久。因此，本文引入跨度评分模型 $g(\cdot)$ 为每个文本跨度评分 $g(s) \in [0,1]$，判断跨度是实体候选的可能性。

该模型依据跨度起止位置的两个字符进行分类，对于任意跨度 $s = (i, j)$，跨度评分的过程可以表示为：

$$(\boldsymbol{h}_1, \dots, \boldsymbol{h}_n) = \text{Encoder}(x)$$

$$g(s) = \sigma\left(\text{FFN}\left(\boldsymbol{h}_i \oplus \boldsymbol{h}_j\right)\right)$$

其中，$\boldsymbol{h}_i$ 指为句子 x 中第 i 个词 w_i 由编码器 Encoder 编码得到的表征，σ 表示 sigmoid 函数，$\oplus$ 指表征向量拼接操作，FFN 指前馈神经网络 (Forward-Feeding Network)。在评分模型判断前，模型参考标点进行中文跨度的筛选，即丢弃包含中文分句标点的跨度。这意味着，保留的任意跨度 $s = (i, j)$ 满足：

$$\left(w_i, w_{i+1}, \dots, w_j\right) \cap \{,\ ;\ 。\ ?\ !\ \dots\} = \emptyset$$

在基于跨度的命名实体识别方法中，分类步骤的召回率会由选择步骤的召回率所限制。筛选过程有效地减少了跨度候选的平均长度与数量，更为重要的是跨度评分模型 $g(s)$ 的效果：现有工作指出，对于重视召回的任务，训练过程需要更强的负例[20]以提升模型性能。所以，经由打分函数排序后，选取其得分较高的，数量为 θ_{top-sp} 的跨度，可以有效兼顾“降低候选数量”和“提升模型召回性能”的效果。

2.3 分类模块

在本文提出的模型网络结构中，分类子模块包含字面 (name)、上下文 (context) 和综合 (combination) 三类。不同模块各自依据其获取的表征为跨度预测实体标签。其中，上下文模块着重跨度外部文本中所包含的信息。综合模块则是参考、分析并平衡字面与上下文子模块提供的信息后，得出实体识别的最终结果。对于句子 $x=(w_1,\dots,w_n)$ 中的跨度 $s=(w_i,w_{i+1},\dots,w_j)$，将依次介绍模块的跨度表征分类过程：

字面模块为图 1 中上端路径部分。借助编码器获取文本表征后，模型将提及首尾位置的字表征拼接以获取字面表征，实现中利用 BERT[7]来为字面模块提供语义特征：

$$(\boldsymbol{h}_1,\dots,\boldsymbol{h}_n)=\text{Encoder}(x)$$

$$\boldsymbol{h}_{name}(s)=\boldsymbol{h}_i\oplus\boldsymbol{h}_j$$

参考现有基于跨度的 NER 方法中常用的表征拼接[15]，$\boldsymbol{h}_i$ 是编码器 Encoder 获取得到的字粒度表征，$\oplus$ 为向量拼接操作。值得注意的是字面表征的信息来源于完整句子，而非仅涉及字面文本，字面模块本质是可感知上下文信息的子模型，但训练其“专注于”字面信息，相应的上下文模块“只感知”上下文信息。严格与上下文解耦的字面模块本质与实体词库作用相同，因此没有必要因此为字面模块的表征能力降档。

上下文模块为图 1 中下端路径部分。网络设计中将独立建模上下文模块，杜绝模型“借助字面信息”捷径学习的可能。模型将句子 x 的字嵌入送入双向 LSTM 编码器，获取每个字符两个方向的文本表征：

$$(\boldsymbol{h}_1,\dots,\boldsymbol{h}_n)=\text{BiLSTM}(x)$$

字表征包含了正、反两个方向的表征 $\boldsymbol{h}_i=[\overrightarrow{\boldsymbol{h}_i};\overleftarrow{\boldsymbol{h}_i}]$，本文基于注意力机制[21]设计了提及上下文边界表征：

$$\boldsymbol{q}=\boldsymbol{h}'_{ctx}(s)=[\overrightarrow{\boldsymbol{h}_{\iota-l}},\overleftarrow{\boldsymbol{h}_{J+1}}]$$

$$\boldsymbol{K}=\boldsymbol{V}=[\overrightarrow{\boldsymbol{h}_1},\dots,\overrightarrow{\boldsymbol{h}_{\iota-1}},\overleftarrow{\boldsymbol{h}_{J+1}},\dots,\overleftarrow{\boldsymbol{h}_n}]$$

$$\boldsymbol{h}_{ctx}(s)=\text{Softmax}\left(\boldsymbol{q}\cdot\boldsymbol{K}^T/\sqrt{n_h}\right)\cdot\boldsymbol{V}$$

其中，n_h 为预设的表征向量维数。在上下文模块中，只有止于跨度起始位置之前的正向编码表征会被该模块所用。同理，反向编码的表征也仅有跨度终止位置之后字符的部分会被使用。至此，任何编码环节中都不涉及字面文本，保证了上下文模块的独立编码。自注意力机制则用于学习哪些字符对判断更加重要。

本文提出的上下文建模方式仅需对原句进行单次编码，即可同步并行计算所有提及表征。可并行、高复用的设计可为高效生成上下文表征提供支持。现有方法通过特殊字符覆盖字面[2]，或在字面左右插入位置标识符号[16]，都因需要为每个跨度重新生成全句文本表征而花费数百倍的时间。

综合模块为图 1 右侧的虚线框部分。模型末端的综合模块通过融合字面模块得出的表征 $\boldsymbol{h}_{name}$ 及上下文模块中得出的表征 $\boldsymbol{h}_{ctx}$，得到综合表征 $\boldsymbol{h}(s)$ 用于模型的最终判断。

$$\boldsymbol{h}_{nc}(s)=\alpha\cdot\boldsymbol{h}_{name}(s)+(1-\alpha)\cdot\boldsymbol{h}_{ctx}(s)$$

$$\boldsymbol{h}(s)=\boldsymbol{h}_{nc}(s)\oplus emb_{\text{width}}[|s|]$$

其中，emb_{width} 是可训练的长度表征 (width embedding)。$\alpha\in[0,1]$ 是综合模块中的平衡参数，值越趋近 1，结果越取决于字面模块，反之，当难以借助字面信息判断实体标签时，平衡参数将引导模型更多地依赖

上下文信息。当两个模块正确地关注相同的特征，模型仍可有效地在平衡参数任意取值下做出正确判断。模型中的平衡参数通过双仿射 (Bi-affine) 的方式从字面、上下文表征中计算获取：

$$\alpha = \text{Sigmoid}\left(\boldsymbol{h}_{name}^{T}(s) \cdot \boldsymbol{W}_{\text{biaf}} \cdot \boldsymbol{h}_{ctx}(s)\right)$$

其中，$\boldsymbol{W}_{\text{biaf}}$ 是双仿射中的可学习参数，用于学习判断当前跨度的标签预测是否需要借助上下文的信息。

至此，综合模块获得了最终的跨度表征 $\boldsymbol{h}(s)$，与其它子模块一样，表征送入 FFN 层后通过 Softmax 函数以获取在模型所有标签上的预测概率分布 $p(s)$。特别地，当多个正例跨度重叠时，根据跨度预测概率仅保留相互重叠跨度中 $max\left(p(s)\right)$ 得分最高的一个。对于概率分布的学习，训练使用交叉熵作为损失函数：

$$\mathcal{L}_{com} = \sum_{s \in S} \sum_{l \in L} \left(-y(s)^{(l)} \log\left(p(s)^{(l)}\right)\right)$$

对于有监督标签的样本，模型借助基于负对数似然 (negative log-likelihood) 的损失函数学习平衡参数 α，综合模块的损失函数为 $\mathcal{L} = \mathcal{L}_{\alpha} + \mathcal{L}_{com}$。

3　数据增强

新增的上下文模块和平衡参数需要标注信息指导训练，但已有标注数据集中不具备为这些新增模块提供监督的标签。已有数据标签无法扩展为上下文与综合模块的正负例完备样本，仅依赖无监督训练很难保证模型学到的平衡参数不会因过偏而失去有效性。面对这一挑战，本文提出了数据增强方法，基于已有的实体标注数据集，针对各个模块生成标签完备的增强样本，应用于模型的增强样本训练与不变性测试中。

3.1　增强策略设计

已有数据增强研究[19]通过替换原句中的实体为其它语料或语种的同类型实体 (即表 2 中的“提及替换”数据增强)，提升了模型的泛化能力。但是，此类数据增强方法无法提供训练平衡参数所需的标签。

表 2 数据增强策略

数据增强策略	转换后的样本（句子）	字面标签	上下文标签
提及替换	史密斯 阿姨吃了这个蛋糕	√	√
随机提及替换	饮用 阿姨吃了这个蛋糕	×	√
跨度边界偏移	凯特阿 姨吃了这个蛋糕	×	×
提及插入	我们吃了这个蛋 史密斯 糕.	√	×
随机提及插入	我们吃了这个蛋 饮用 糕	×	×

表 2 中展示了本文中的数据增强策略，其中重点设计了两类字面标签与上下文标签相反的特殊策略：**随机提及替换**和**提及插入**。这两类数据增强策略仅需要收集任意已有数据集中的标注实体与非实体词组，与不同的上下文之间进行组合，从而产生与所使用数据集语料相同的样本集合，降低了数据收集需求。增强方法会构成令两个子模块标签相反的样本，并为平衡参数设置标签，这样的设计用于训练模型：1) 在仅凭字面难以判断的情况下，应转而依赖上下文来进行判断；2) 鼓励模型从未知的上下文中提取确定的实体。

这两类策略产生的样本会存在语法、句法问题，但模型末端的分类器不会从中进行学习，这保证了这两类样本仅用于字面、上下文模块的训练中，并指导模型对平衡参数的学习。为各模块标签的完备性考虑，本文也分别为几个模块设计了负例标签的数据增强策略，同时用于生成与真实实体边界接近的负例。至此，通过数据增强方法，使每个模块都构造出相应的正例、负例标签，将额外的子模块训练从无监督转变为弱监督学习，以期不同模块能够各司其职地掌握其判别能力。

3.2 训练数据增强

标签设置 (对于标签 t)	存在标签 t 实体的样本			不存在标签 t 实体的样本		标签完备性
	替换类策略			插入类策略		
	提及替换 50%概率	随机提及替换 50%概率	跨度边界偏移 (同时应用)	提及插入 50%概率	随机提及插入 50%概率	
字面	☑	☒	☒	☑	☒	☑☒
上下文	☑	☑	☒	☒	☒	☑☒
综合标签	☑	-	☒	-	-	☑☒
平衡参数 α	-	Context	-	Name	-	Context Name

图 2 数据增强策略的标签完备性

数据增强旨在为模型各模块生成伪标签数据，特别是字面模块与上下文模块标签不同的样本。此类样本将平衡参数的学习从无监督转为弱监督。基于前文所述的数据增强策略，图 2 中展示了不同策略的标签设置，其中 ☑ 指**保持原有标签** $\hat{y}(s)$，即标签为该跨度原本的实体类型；☒ 指**设置负例标签** (即 None，“非实体”标签)；留空的位置指忽略该模块的损失函数计算，不为该模块计算 loss 用于反向传播学习。

其中，标有“替换类策略”的方框中展示了替换类的两种增强策略。给定句子 x 与其中一个特定实体跨度 $s = (w_i, w_{i+1}, \dots, w_j)$, $\hat{y}(s) \in \mathcal{E}$，将跨度 s 替换为另一个字符序列 $s' = (w'_1, \dots, w'_m)$，对于提及替换，s' 是与 s 具有相同实体标签 $\hat{y}(s) \in \mathcal{E}$ 的其它提及，从而得到新序列 $x' = (w_1, \dots, w_{i-1}, w'_1, \dots, w'_m, w_{j+1}, \dots, w_n)$。对于随机提及替换，$s'$ 为同语料的随机字串。Dai 等人[22]的工作中也曾用到过与上述提及替换策略相似的方法，但并未带来提升，这可能是因为仅有此类增强策略还并不够完备。

另外，标有“插入类策略”的方框中展示了插入类的两种策略。该策略会在句子 x 中随机位置插入文本片段 $s' = (w'_1, \dots, w'_m)$，得到新序列 $x' = (w_1, \dots, w_{i-1}, w'_1, \dots, w'_m, w_i, \dots, w_n)$。对于提及插入策略而言，$s'$ 是从同数据集中具有实体标签 $\hat{y}(s') \in \mathcal{E}$ 的其它提及。

对于平衡参数 α 的标签设置参考图 2 的最后一行。在随机提及替换策略中，平衡参数的标签设置为依赖上下文信息，用于当模型未在数据集中见过跨度字面的文本时，鼓励模型借助上下文信息预测实体。而在随机提及插入策略中，平衡参数的标签设置为偏向字面信息，即当确定的字面处于不确定的上下文中时，指导模型主要借助字面信息作出判断。综上所述，相反标签样本指导模型学习关注不同来源的特征。

这些策略普遍令人在意的是会产生混乱、不合理的样本，因为无法为其分配合适的实体标签，这样的样本难以纳入大多数现有模型。然而，本文中它们仅用于指导模型应按怎样的比例从模块获取信息，此时模拟了极端情况的字面、上下文搭配，在其它模型训练或评估的场景下不会利用这些反直觉的构造样本。

3.3 测试数据增强

作为 CheckList 中 Invariance Test 的扩展，本文提出一种对 NER 模型的评估方法及相应指标用于度量 NER 模型的不变性泛化能力，其本质是借助已有测试集生成一批新的测试集以用于评测模型性能。新测试集中的每个样本通过同标签提及替换的方式得到，而同标签实体通常具有句法结构上的等位性，保证了所得样本的语法正确。首先，从测试集中随机抽取 n_{it} 个句子，对于预定义实体类型中的每个实体 $t \in \mathcal{E}_{it}$，提取所有在抽取的句子中出现过的提及 M。对于含有 t 类型的提及的所有句子 X，用 M 中的提及分别替换这个具有类型 t 的实体提及。在这个过程中，会产生数量为 $|M| \times |X|$ 的构造提及样本，待测试模型的目标为从修改后的样本句子中，成功召回并预测出这些构造样本中的实体提及和正确的实体标签。

4 实验设计和结果分析

4.1 数据集

本文中使用的中文 NER 数据集包括：参照 lemonhu/NER-BERT-pytorch 的切割方式，将 **MSRA** (Microsoft Research Asia[23]) 训练集拆分部分作为验证集；Zhang and Yang[11] 提出源于中文简历的 **Resume** 数据集；以及 Weischedel 等人[24]提供的 OntoNotes 系列数据集，选择中文版本 **Onto4** 数据集用于测试。

4.2 基线模型

本文选取以下基线模型用于对比：将中文 NER 中最常见的 **LSTM-CRF**[6] 和 **BERT-CRF** 方法在本文中作为 NER 基线模型。为公平对比，本文以这两个基础编码框架作为编码模块的主要组件，不借助条件随机场的情况下设计了 NCModel。此外，借助外部词汇晶格编码 (**Lattice-based**) 的 LatticeLSTM[11]、Soft-Lexicon[10] 和 DCSAN[1] 等模型先后成为了最优的中文命名实体识别方法。而在现有的英文命名实体识别模型中，具有最优表现的是基于跨度 (**Span-based**) 的模型。Wadden 等人[15]与 Zhong 等人[16] 提出的表征方式在不引入额外知识的情况下，于多个英文实体数据集上取得了最优效果。受此启发，本文将其应用于中文实体识别模型中，在 NCModel 里作为字粒度的基本表征。

4.3 实验设定

为保证对比实验公平性，本文中细致地考虑了表征选择、超参设置以及评测指标的设计，具体的模型与配置源码已发布于公开代码库 (anonymous.4open.science/r/CDNER-for-IPM22) 中。

4.3.1 表征选择

LSTM 网络中采用的字嵌入来源于 BERT 模型 (版本为 bert_chinese_L-12_H-768_A-12) 的预训练嵌入参数。这意味着在对比中使用了相同的字嵌入获取方法，包括位置表征 (position embeddings) 和层正则化 (layer normalization)。在 NCModel 中，字面模块和上下文模块共享 BERT Encoder 中的字嵌入层参数。

4.3.2　超参设置

训练时，对于所有的中文命名实体数据集，采用相同设置：每步训练使用 4 个样本句子 (batch size)，跨度长度限制 (l_{sp}) 为 25，且均在训练 10 轮 (training epoch count) 后取验证集效果最好的模型。预训练模型 BERT 内部参数的学习率设置为 1e-5，其余各模块参数学习率设为 5e-4。关于在第 2.2 节中提到的跨度选择，简单地通过步长为 50 的搜索 (grid-search) 方法，确定了在单张 12G 2080Ti 显卡上不产生显存超限 (Out-Of-Memory, OOM) 的最大超参 $\theta_{top-sp} = 300$。经验证，此时在所有中文数据集上的实体召回率均不低于 98%。

4.3.3　评测指标

本文使用常见的 F1-Score 作为评测指标。参考 Xiao 等人[25]提出的严格验证方式，当且仅当跨度 (l, r) 的边界和实体类型 t 同时正确时为一处正确的实体预测，在跨度选择阶段丢失的实体跨度会被视作未召回。

另外，本文中还提出了不变性测试的失败率 (Failure Rate, **FR**) 指标，即模型标签判断失败的占比，用于度量模型在第不变性测试中体现的泛化能力。对于构造获得的跨度集合 S，失败率的计算为：

$$\boldsymbol{FR} = \frac{1}{|S|}\sum_{s_i \in S} sgn\big(\hat{y}(s_i) \neq y(s_i)\big)$$

其中，$sgn(\cdot)$ 为符号函数，当其内部条件为真时取值为 1，反之为 0。

4.4　实验结果

4.4.1　不变性实验

表 3 不变性能力测试中各模型 FR 指标对比

模型	BiLSTM-CRF	LatticeLSTM	BERT-CRF		Span-based	NCModel (Ours)
			初始版本	加入数据增强		
人名实体	14.5	6.7	3.4	1.9	2.3	**1.6**
公司名实体	25.1	23.4	14.5	14.6	14.3	**9.3**
综合统计	19.0	13.6	8.1	7.3	7.4	**4.8**

基于 3.3 节的方式构建样本设计实验，检测模型在不变性测试用例上的性能评估泛化能力。以 MSRA 数据集为例，设置目标标签 $\mathcal{E}_{it}$ ={人名, 公司名}，表 3 中我们提出的方法在 FR 上显著优于其它基线框架。

不变性实验构建测试集时，需要保证同类实体相互替换后原文语义依然正确，部分数据集中，“地址” (xx 路 xx 号) 与 “地理位置” (xx 省 xx 市) 统一标注为 “地名”，为保证替换后不出现语法错误，测试选择了对于 NER 而言最为简单常见的人名与公司名实体，然而表中所有模型的综合错误率都明显高于原测试集。可以看出，较之原测试集的准确率测试，不变性测试对于现有的实体识别模型而言是一种更难的评测方式。

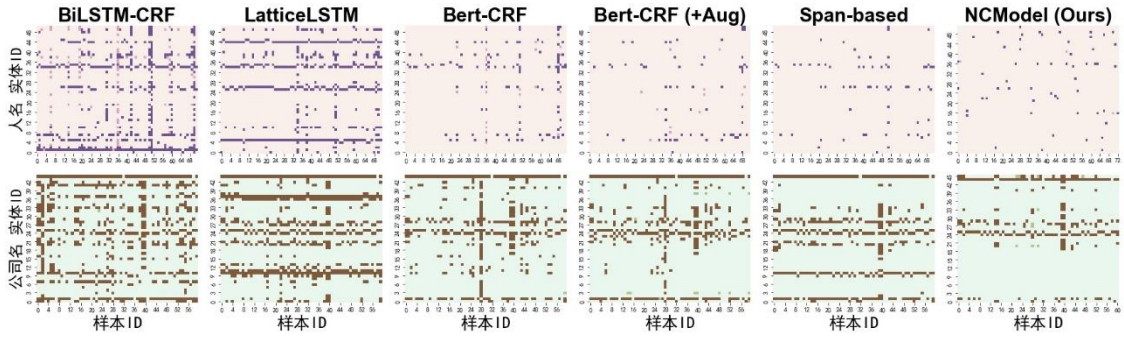

图 3　在 MSRA 数据集上，各模型框架的可视化不变性测试结果

为深入讨论本文提出的方法如何提升模型的不变性能力，绘制了如图 3 所示的热力图以展示具体的错误率分布。结果中，每一列为代表不同模型，第一行的热力图表示人名实体的测试，第二行为公司名。每张热力图中横轴为样本编号，理解为当前样本点所使用的上下文，纵轴为实体编号，理解为当前样本点所使用的实体提及文本，召回失败时为该点着色。指标 FR 也可解释为热力图中着色点所占的比例，占比越少表示模型不变性泛化能力越好，在扰动中维持预测结果的能力越强。当热力图的某行出现较多的着色点，意味着对模型而言，该行对应的字面较难判断。同理，竖线状图案表示模型难以根据该上下文判断实体。

可以发现，NER 模型更多地因字面导致召回失败，侧面印证了模型借助上下文信息的必要性。本文提出的模型明显减少了热力图中的横线状图案，为 BERT-CRF 模型引入数据增强方法 (表示为 +Aug) 后降低了 FR，应用提出的模型网络将进一步降低 2.3%的 FR，获得了 30%以上的相对提升。在不变性能力对比中模型排序为：NCModel > BERT-CRF (+Aug) ≈ Span-based ≈ BERT-CRF > Lattice-LSTM > BiLSTM-CRF。现有模型的性能仍受限于上下文模型表征能力，可以期待更好的上下文表征方法应用于该编码框架上的效果。

本文进一步地从平衡参数角度入手对模型表现提升的原因进行了分析。从 MSRA 数据集的测试集中构造新的增强数据集，分析了模型预测时平衡参数的表现与分布情况：

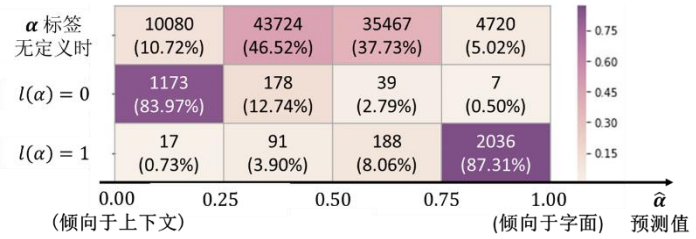

图 4　NCModel 模型中平衡参数的预测分布

使用在 MSRA 训练集上训练得到的模型，在增强数据集上预测得到图 4 所示的 α 值分布。可以看到，当构造标签为 0，即应倾向于上下文时，模型预测中平衡参数有 96% 的比例取值趋近于 0，即令模型更加倾向于借助上下文进行预测，其中 84% 的样本接近于 0。同时，当构造标签为 1，即应倾向于字面表征用于预测时，模型也有 95% 以上的比例给出正确的平衡参数，指导模型依赖字面表征用于判断。在未定义平衡参数标签的原始样本中，平衡参数不会过偏于某一侧，而是处于 0.5 附近以较为中立的态度指导模型。该分布揭示了仅借助少量的伪标签增强，即可成功训练出为现有模型提供不变性泛化能力提升的编码框架。

综上，本文提出的方法在不变性测试中表现优秀，可有效提升中文 NER 模型重要的泛化能力。人类能够做到在相同上下文中举一反三地判断实体，独立建模上下文模块也可为模型更好地学习掌握这一能力。

4.4.2　消融实验

本节通过消融实验量化各组件为性能提升发挥的作用，结果如表 4 所示：多数情况下，缺少任一类增强策略都会令模型的实体识别效果与不变性泛化能力下降。这符合本文中提出的观点：模型需要完备的数据增强策略，当增强数据能够为不同模块提供完备的伪标签时，才能有效辅助模型中每个模块的训练。另外，为已有成熟模型引入独立建模的上下文模块，也并未影响原有模型的预测性能；上下文模块缺少了对于字面文本的编码，难以独自完成 NER 任务，但借助本文提出的模型框架仍可为模型带来提升。在模型网

络中添加上下文模块的主要目的正是在“仅凭字面含义不足以判断”甚至产生混淆时，令模型能够借助上下文来做出判断，从而达到与字面模块互补的效果。

表 4 增强策略消融结果

数据集	MSRA		Resume		Onto4	
	F1-Score	FR	F1-Score	FR	F1-Score	FR
NCModel	**95.4**	4.83	**96.3**	**2.73**	**81.0**	21.79
- 替换类策略	95.0	5.14	96.2	2.85	79.9	22.54
- 插入类策略	95.2	**4.74**	95.8	3.39	80.5	24.08
- 上下文模块 (Span-based)	95.3	5.03	95.9	2.94	80.8	**21.00**
- 上下文模块 (BERT-CRF)	94.1	-	94.6	-	80.4	-
- 字面模块 (Span-based)	42.7	53.48	67.4	35.12	17.1	88.51

4.4.3 实体识别效果

实验使用 F1-Score 评估在不同框架下模型的整体实体识别能力，表 5 中展示了在各数据集上的表现。将在 MSRA 数据集上的表现与表 3 中不变性测试的失败率比对可以发现，较之准确率而言，模型在不变性测试中通常具有更高的失败率，这印证了目前 NER 模型在不变性能力上有所不足；而与表 4 横向对照可以发现，引入独立建模的上下文模块后，模型在不变性能力上的失败率已经十分接近综合准确率的表现，这说明该方法可以消除不变性泛化能力在现有模型中的短板效应。

表 6 中文 NER 模型基本框架的实体识别性能

数据集	模型	准确率	召回率	F1-Score
MSRA	Bi-LSTM CRF [6]	93.0	90.8	91.9
	LatticeLSTM [11]	94.0	92.2	93.1
	BERT-CRF [7]	95.0	93.3	94.1
	Span-based [16]	95.6	95.3	**95.4**
	NCModel (Ours)	95.7	95.0	**95.4**
Resume	Bi-LSTM CRF	94.5	94.3	94.4
	LatticeLSTM	94.8	94.1	94.5
	BERT-CRF	94.0	95.2	94.6
	Span-based	95.5	96.0	95.8
	NCModel (Ours)	96.2	96.5	**96.3**
Onto4	Bi-LSTM CRF	74.4	69.4	71.8
	LatticeLSTM	74.2	73.1	73.6
	BERT-CRF	82.4	78.4	80.4
	Span-based	83.5	78.5	80.9
	NCModel (Ours)	80.9	81.1	**81.0**

本文着眼于提出从不同角度为跨度进行编码的方式，仅利用基本的 BERT 及 LSTM 编码方式，不借助外部特征和工具，为模型从不变性的角度提升泛化能力。令人振奋的是，其综合表现并未因引入表征能力较差的上下文表征而受到影响，与现有 NER 解决方案相当，较之 BERT-CRF，准确率平均提高了 1.2%。

5 结论

本文提出了一个同时包含字面模块和上下文模块的 NER 模型，设置平衡参数用于指导模型分类时是否应当更加依赖上下文信息。该方法借助专门设计的数据增强手段，为各模块提供无法从现有数据集中获得的伪标签，从而帮助模型同时学习平衡参数与各个模块的表征。实验显示，提出的模型可以在保证准确率的同时，在不变性能力测试中带来较大提升，也验证了该方法有助于帮助模型提升泛化能力。通过一系列实验印证了现有模型可以借助上下文信息来弥补模型在预测未知字面时的不足。

参考文献

[1] Zhao S, Hu M, Cai Z, et al. Dynamic modeling cross- and self-lattice attention network for chinese ner [C]// Thirty-Fifth AAAI Conference on Artificial Intelligence, Virtual Event, 2021: 14515-14523.

[2] Gui T, Wang X, Zhang Q, et al. TextFlint: Unified Multilingual Robustness Evaluation Toolkit for Natural Language Processing [C]// Proceedings of the Joint Conference of the 59th Annual Meeting of the Association for Computational Linguistics, Online, 2021: 347-355.

[3] Field A, Tsvetkov Y. Entity-Centric Contextual Affective Analysis [C]// Annual Meeting of the Association for Computational Linguistics, Florence, Italy, 2019: 1462-1467.

[4] Agarwal O, Yang Y, Wallace B C, et al. Interpretability Analysis for Named Entity Recognition to Understand System Predictions and How They Can Improve [J]. Comput. Linguistics, 2021: 117-140.

[5] Ribeiro M T, Wu T, Guestrin C, et al. Beyond Accuracy: Behavioral Testing of NLP Models with CheckList [C]// Annual Meeting of the Association for Computational Linguistics, Online, 2020: 4902-4912.

[6] Huang Z, Xu W, Yu K. Bidirectional LSTM-CRF Models for Sequence Tagging [J/OL]. *ArXiV*, 2015.

[7] Devlin J, Chang M W, Lee K, et al. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding [C]// Conference of the North American Chapter of the Association for Computational Linguistics, Minneapolis, USA, 2019: 4171-4186.

[8] Fu J, Liu P, Zhang Q. Rethinking Generalization of Neural Models: A Named Entity Recognition Case Study [C]// The Thirty-Fourth AAAI Conference on Artificial Intelligence, New York, NY, USA, 2020: 7732-7739.

[9] Cao Y, Chen D, Li H, et al. Nested Relation Extraction with Iterative Neural Network [C]// Proceedings of the 28th ACM International Conference on Information and Knowledge Management, Beijing, China, 2019: 1001-1010.

[10] Ma R, Peng M, Zhang Q, et al. Simplify the usage of lexicon in chinese ner [C]// Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, Online, 2020: 5951-5960.

[11] Zhang Y, Yang J. Chinese NER Using Lattice LSTM [C]// Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics, Melbourne, Australia, 2018: 1554-1564.

[12] Xuan Z, Bao R, Jiang S. FGN: Fusion Glyph Network for Chinese Named Entity Recognition [C]// Knowledge Graph and Semantic Computing: Knowledge Graph and Cognitive Intelligence - 5th China Conference, Nanchang, China, 2020: 28-40.

[13] Zhu E, Li J. Boundary smoothing for named entity recognition [C]// Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics, Dublin, Ireland, 2022: 7096-7108.

[14] Li J, Fei H, Liu J, et al. Unified named entity recognition as word-word relation classification [C]// Thirty-Sixth AAAI Conference on Artificial Intelligence, Virtual Event, 2022: 10965-10973.

[15] Wadden D, Wennberg U, Luan Y, et al. Entity, Relation, and Event Extraction with Contextualized Span Representations [C]// Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing, Hong Kong, China, 2019: 5783-5788.

[16] Zhong Z, Chen D. A frustratingly easy approach for entity and relation extraction [C]// Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Online, 2021.

[17] Yaghoobzadeh Y, Schütze H. Corpus-level Fine-grained Entity Typing Using Contextual Information [C]// Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing, Lisbon, Portugal, 2015: 715-725.

[18] Lin B Y, Lee D H, Shen M, et al. Trigger NER: Learning with Entity Triggers as Explanations for Named Entity Recognition [C]// Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, Online, 2020: 8503-8511.

[19] Agarwal O, Wallace B C, Yang Y, et al. Entity-switched datasets: An approach to auditing the in-domain robustness of named entity recognition models [J]. ArXiv, 2020.

[20] Eberts M, Ulges A. Span-based Joint Entity and Relation Extraction with Transformer Pre-training [J]. *Frontiers in Artificial Intelligence and Applications*, 2019(325): 2006-2013.

[21] Vaswani A, Shazeer N, Parmar N, et al. Attention is all you need [C]// Annual Conference on Neural Information Processing Systems 2017, Long Beach, CA, USA, 2017: 5998-6008.

[22] Dai X, Adel H. An Analysis of Simple Data Augmentation for Named Entity Recognition [C]// Proceedings of the 28th International Conference on Computational Linguistics, Barcelona, Spain, 2020: 3861-3867.

[23] Levow G A. The third international chinese language processing bakeoff: Word segmentation and named entity recognition. [M]// In Proceedings of the Fifth SIGHAN Workshop on Chinese Language Processing. Sydney, Australia, 2006: 108-117.

[24] Weischedel R, Palmer M, Marcus M, et al. OntoNotes Release 4.0 LDC2011T03 [EB/OL]. 2011.

[25] Xiao S, Ouyang Y, Rong W, et al. Similarity Based Auxiliary Classifier for Named Entity Recognition [C]// Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing, Hong Kong, China 2019: 1140-1149.

Explicitly Modeling the Context for Chinese Named Entity Recognition

Dian Chen* Yixuan Cao* Ping Luo*△

(* Key Lab of Intelligent Information Processing of Chinese Academy of Sciences (CAS),
Institute of Computing Technology, CAS, Beijing 100190, China)
(* University of Chinese Academy of Sciences, Beijing 100049, China)
(△ PengCheng Lab, Shenzhen 518066)

Abstract

Current Chinese NER models have achieved remarkable results on public datasets. However, some studies suggest that they rely too heavily on literal features of entity text. Moreover, the influence of context on entity recognition has yet to be fully explored. As a result, it leads to poor performance in simple "invariance tests," such as replacing a company name with another one in the same context, resulting in an incorrect prediction. To address this issue, we propose independently modeling the contexts, allowing the model to distinguish between entity literals and it. Additionally, we introduce an adapted data enhancement method to train the context, surface name, and combination modules. Our approach significantly improves the model's performance in the invariance test without sacrificing recognition performance, reducing the failure rate by 2.3% compared to the benchmark model.

Keywords: natural language processing, Chinese named-entity recognition, explicit modeling, data augmentation