# 基于深度学习的命名实体识别综述

## 命名实体识别任务

### 1.1 命名实体识别概念

在文本中对实体的引用可以有三种形式：命名性指称、名称性指称、代词性指称。例如在句子“**中国乒乓球男队主教练刘国梁出席了会议，他布置了备战世乒赛的具体安排**”中“刘国梁”的指称项有三个，其中“中国乒乓球男队主教练”是名词性指称，“刘国梁”是命名性指称，“他”是代词性指称。

命名实体识别任务是识别出文本中实体的命名性指称项，并标明其类别。一般来说，命名实体识别任务中，实体类别一般分为三大类（实体类、时间类、数字类）、七小类（人名、机构名、地名、时间、日期、货币和百分比）。不同任务对命名实体类别粒度的需求不同，例如在有些任务中只要识别出一个实体是人，有些任务则需要识别出一个实体是教师、学生或者医生。

### 命名实体识别标注方法

目前主流的命名实体标注方法分为两种，一种是BIO方法,一种是BIOES方法。BIO方法中B标注命名实体的开始词、I标注命名实体识别的中间词、O标注不属于命名实体的其他词。BIOES方法中B、I、O的意义与BIO标注方法中的相同，但增加了E和S，E标注命名实体的结束词、S标注只包含一个词的命名实体。通常在标注的过程中，会加上命名实体的类别信息。如下是分别使用BIO方法和BIOES方法对同一个句子进行标注的结果。

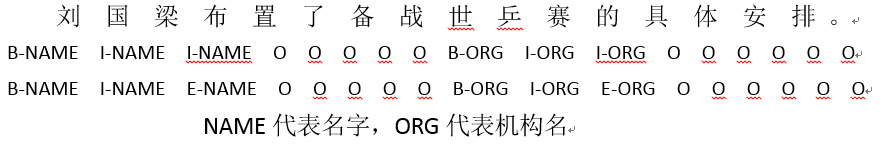


图1.1

### 命名实体识别评估标准

在命名实体识别任务中，只有当预测的命名实体边界和类别与实际完全相同时，才可以认为识别成功。准确率、召回率，F1值是评估命名实体识别系统性能的三大指标。式（1-1）、式（1-2）、式（1-3）分别是它们的计算公式。其中TP(True Positive)表示预测的实体中，正确实体的数量、FP(False Positive)表示预测的实体中错误实体的数量。FN(False Negative)表示语料库中未被预测出的实体数量。

 式（1-1）

 式（1-2）

 式（1-3）

## 基于深度学习的命名实体识别方法

传统的命名实体识别方法有基于规则的命名实体识别方法、基于机器学习的命名实体识别的方法。以上两种都需要依靠大量的人力去构建模板或者特征，而基于深度学习的方法 ，依靠深度学习强大的非线性表达能力，不需要人力去构建模板或者特征，可以实现端到端的模式【】，成为当前命名实体识别方法的主流。

基于深度学习的命名实体识别方法，都遵循着通用的框架，如图2.1所示。通用框架可以分为三个串行的模块，分别是输入的分布式表示、内容编码、标签解码。输入的分布式表示指的是单词、字符的嵌入还有词性、外部知识库的分布式表示。内容编码是通过使用CNN、RNN或者的其他的神经网络获得内容之间的依赖信息。标签解码是预测对应词的标签。

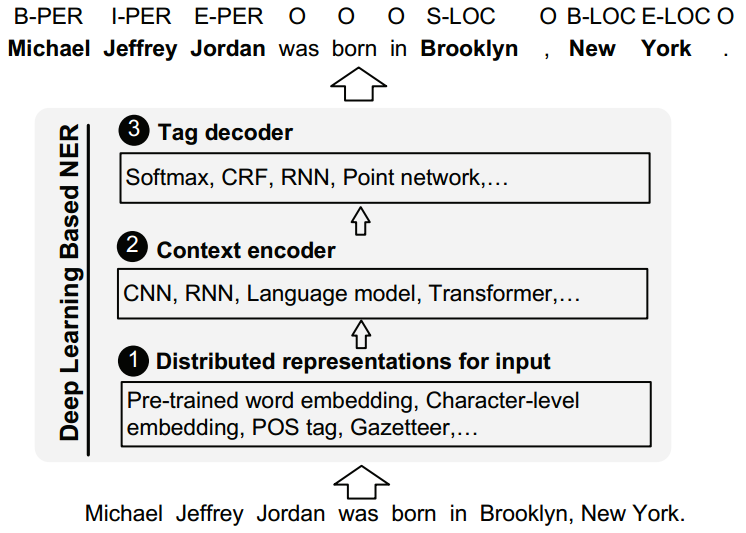


图2.1

### 2.1输入的分布式表示

基于深度学习的命名实体识别方法的输入一般是低维稠密的实值向量，所以需要获得输入的分布式表示。输入的分布式表示一般包括词嵌入、字嵌入，同时一些方法也会考虑词的词性信息、外部知识库wikipedia、gazetteers等信息的嵌入。

#### 2.1.1词嵌入

词嵌入主要分为两类，分别是基于词向量的固定表征包括word2vec、fastText、glove，基于词向量的动态表征包括elmo、GPT、Bert。

word2vec是Mikolov于2013年提出的词嵌入方法，包括skip-gram模型和CBOW模型。

skip-gram模型的原理是给定输入词来预测上下文。CBOW模型则相反是给定上下文来预测输入词。图2.2是两种模型的网络结构。

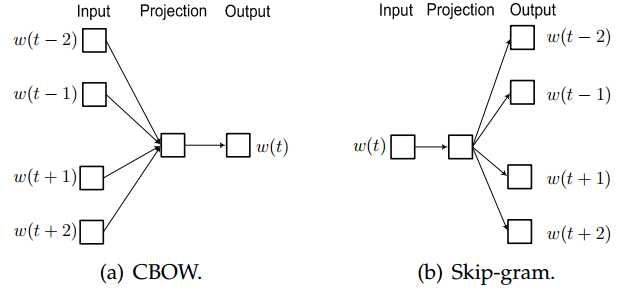


图2.2

Glove【】是Jeffrey与2014年提出的词嵌入方法，是一种半统计半机器学习的方法。与word2vec方法仅考虑输入词附近的窗口信息不同，Glove通过构建词与词之间的窗口共现矩阵考虑了全局的信息。

FastText【】是Mikolov于2016年提出的，在skip-gram基础上考虑词的内部结构信息。

FastText不仅对词本身编码也对其字符级的n-gram特征进行编码，语料中的低频词，可能与某些高频词具有相同的n-gram特征，所以可以得到可靠的向量表示。

Elmo是AllenAI实验室于2018年，提出的一种基于词向量的动态表征方法。word2vec、glove、fasttext都是基于词向量的静态表征方法，每一个词都只有一个固定的向量与之对应。

然而多义词在不同语境中有不同的意思，难以用一个词向量对其所有意思进行编码。所以AllenAl实验室，提出了Elmo方法，在该方法中，一个词的词向量不是固定的，而是根据上下文语境信息动态确定。

GPT、Bert同样也是基于词向量的动态表征方法。与Elmo不同的是，它们都利用Transformer而不是LSTM来提取上下文依赖信息(实践表明, Transformer的特征提取能力要强于LSTM)。GPT与Bert之间的差异性主要体现在GPT采用的是单向语言模型，而Bert采用的是双向语言模型。

#### 2.1.2 字符嵌入

相关研究证明，在英文中，考虑词的内部结构信息（前缀、后缀等信息），可以使命名实体识别系统的性能提升。现在主流一般用基于CNN的模型或者基于RNN的模型做字符嵌入。如图2.3所示。是两种模型的网络结构。Lample等提出的基于BiLSTM+CRF的NER方法是使用双向LSTM做字符嵌入。Chiu等提出的基于LSTM-CNNs的命名实体识别方法是使用CNN做字符嵌入。同样Elmo方法中，也是用CNN提取词的字符级特征。

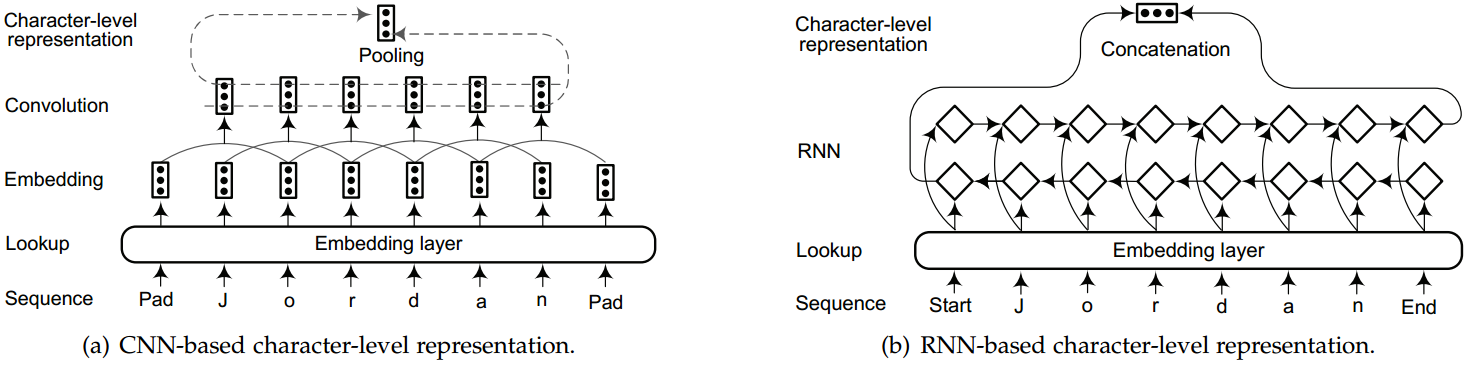


图2.3

### 内容编码

获得输入的分布式表示后，下一阶段就是内容编码，也就是获取上下文依赖信息。

普遍使用的内容编码结构，主要是卷积神经网络、循环神经网络、递归神经网络和最近提出的transformer.

#### 2.2.1卷积神经网络

卷积神经网络是最早用于上下文特征提取的神经网络结构之一，其结构如图2.4所示。依次执行输入序列的分布式表示，卷积、最大池化、全连接，最后得到特征输出。

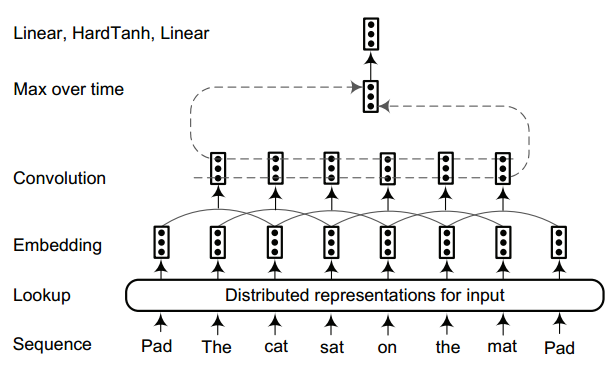


图2.4

#### 2.2.2 循环神经网络

循环神经网络是目前最流行的用于处理序列数据的神经网络框架。其网络结构如图2.5所示。普通的RNN在处理长序列时，可能会出现梯度消失的问题。梯度消失导致无法对长期依赖进行建模。LSTM、GRU是RNN的变体，可以较好得对长期依赖进行建模，克服梯度消失的问题。另外一个缺点是，单向的RNN仅仅对左边的依赖进行建模，语言学常识认为，当前词的意思不仅与左边的语境有关，同时也和右边的语境有关，所以在命名实体识别任务中，一般用双向的循环神经网络对内容进行编码。

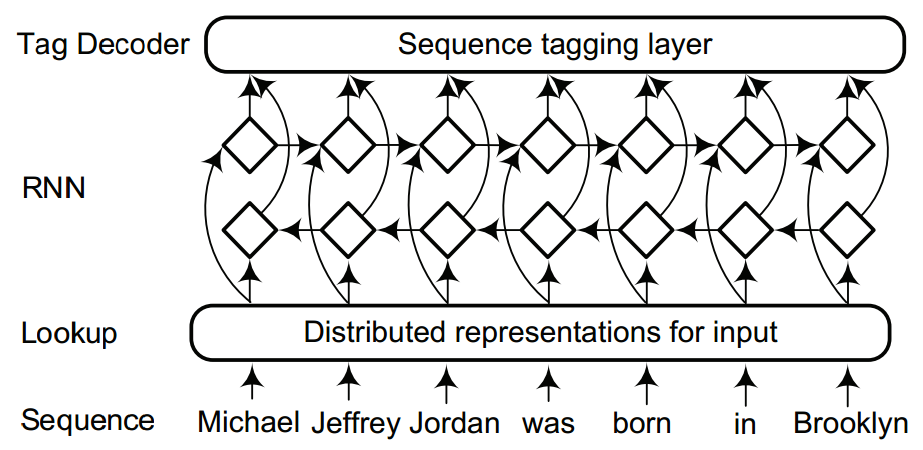


图2.5

#### 2.2.3 递归神经网络

#### 2.2.4 Transformer

Transformer是google在17年提出来的模型。RNN是通过隐藏状态的传递对上下文依赖信息进行编码，Transformer是通过注意力机制（编码当前词对句子中每个单词的注意程度）对上下文依赖信息进行编码。如图2.6所示是其网络结构。相关实验证明Transformer的语义特征提取能力、长距离特征捕获能力、任务综合特征抽取能力、并行计算能力及运行效率均优于循环神经网络

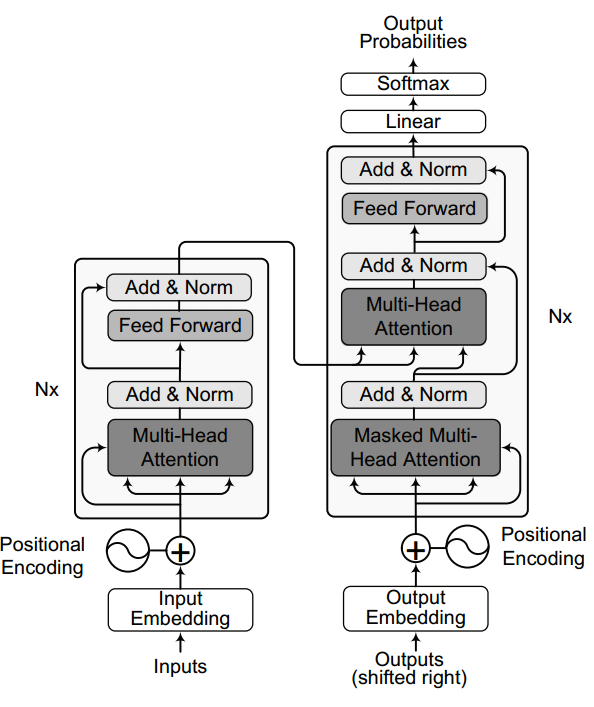


图2.6

### 2.3标签解码

命名实体识别的最后一个阶段，标签解码。标签解码主要的作用是将内容编码阶段的输出映射到相应的标签集，得到每个词的标签。全连接层+softmax层、条件随机场、循环神经网络、指针网络都可以作为标签解码的方法。

#### 2.3.1 全连接层+softmax层

全连接层+softmax层是早期最普遍的标签解码方法。特征向量经过全连接层映射到维度为标签数量的向量，之后经过softmax层得到概率向量，对每个概率向量中最大值对应的标签作为词的标记。softmax计算公式如式（2-1）

 式（2-1）

#### 2.3.2 条件随机场

命名实体识别中，存在序列标注约束，比如B（begin）标签后面不可能接O(other)标签，全连接层+softmax层，难以完全提取序列标注约束信息，可能会出现预测上一个词的标签为B,预测当前词的标签为O。CRF层可以进一步提取序列标注约束信息，提高命名实体识别的准确率。图2.7是其相应的结构。

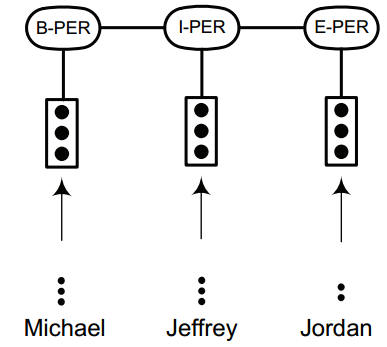


图2.7

#### 2.3.3 循环神经网络

循环神经网络不仅可以用来对内容进行编码，同样也可以用于标签解码。其网络结构如图2.8所示，隐状态的输入为内容编码阶段的输出、上一个词的预测标签、上一个词的隐状态。循环神经网络类似CRF都是为了进一步提取标签约束信息。

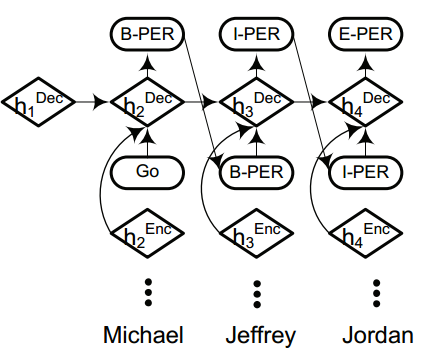


图2.8

#### 2.3.4 指针网络

指针网络的网络结构如图2.9所示。

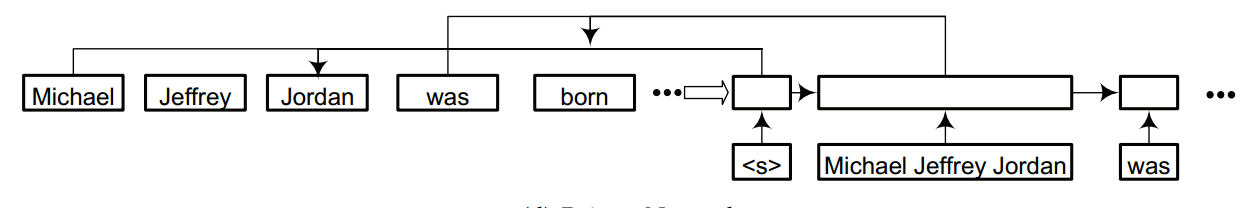


图2.9

## 基于深度学习的命名实体识别的挑战与展望

### 3.1基于深度学习的命名实体识别的挑战

#### 3.1.1 语料标注

基于深度学习的命名实体识别需要耗费人力、物力去标注大量语料。现实中可能存在某种语言的语料不足，或者特定领域的语料需要领域专家来标注，以上两种情况，对基于深度学习的命名实体识别来说，都是巨大的挑战。

语料标注的质量，对实体识别系统的性能也有很大的影响。在目前的语料库中，会存在相当一部分的嵌套实体。嵌套实体，就是实体里面包含着实体。比如“Empire State”、“Emprie state Building”在语料库中都被标记为Location，这样会造成实体的边界有歧义。同时细粒度实体识别的任务中，实体可能会有多个类别。现有的标注方法都难以对嵌套实体、细粒度实体进行标记，所以急需要一套通用的标记模式。

#### 3.1.2 信息文本和未知实体

目前命名实体识别用的语料都是新闻、文章等规整的语料。然而互联网中绝大多数的语料都是不规整的语料，现有的方法在不规整的语料中的F值仅0.4.所以对命名实体识别来说，不规整的文本也是一个挑战。

命名实体识别系统在识别未登录实体（在训练集中未出现的实体）的准确率较低。所以如何提高命名实体识别系统的迁移、泛化能力，也是一个巨大的挑战。

### 基于深度学习的命名实体识别的展望

#### 3.2.1细粒度命名实体识别

目前主流的工作都聚焦于粗粒度命名实体识别，但是传统的粗粒度命名实体识别难以满足工业界的需求，所以粗粒度的命名实体识别同样需要得到关注。细粒度命名实体识别的主要难点在于一个实体可能会有多个类别。可以考虑将细粒度命名实体识别划分为两个子任务，一个子任务用于命名实体的边界确定，另一个子任务对实体进行分类。

#### 3.2.2命名实体识别和实体链接联合学习

实体链接是指将文本中的实体链接到知识库中的相似实体。大多数研究认为实体识别和实体链接是两个独立的串行任务（实体识别后进行实体链接）。但是可以知道实体识别的成功可以提高实体链接的准确率，同样实体链接的成功也可以提高实体识别任务中实体边界、类别预测的成功率。所以可以考虑命名实体识别和实体链接联合学习，一方面可以避免误差的传递，另一方面两个任务也存在互利现象。

#### 3.2.3基于迁移学习的命名实体识别

许多应用程序采用现有的命名实体识别系统来识别命名实体。但是由于输入的语料的语言特点差异，可能导致命名实体识别系统难以工作。所以需要更多的努力致力于基于迁移学习的命名实体识别

## 结论

本文首先介绍了命名实体识别任务、命名实体标注方法、命名实体识别的评估标准。之后按顺序详细介绍了输入的分布式表示、内容编码、标签解码（基于深度学习的命名实体识别方法的三个通用串行框架）。最后我们总结了基于深度学习的命名实体识别面临的挑战并对命名实体识别未来的方向进行了展望。