自然语言处理技术概览：以开放域信息抽取为例分析

石睿（211300024,211300024@smail.nju.edu.cn）

(南京大学 人工智能学院, 南京 210093)

摘　要:　自然语言处理（Natural Language Processing，NLP）作为 AI 的子领域，旨在让计算机理解和使用人类语言从而执行对应任务。同时由于语言作为人类日常生活中的重要沟通介质，NLP 的重要性不言而喻，其广泛被应用于产业界的客服、医疗、金融、教育等领域中。同时 NLP 领域发展迅速，经过二十余年的技术更新已经从基于统计和形态学、语义学的研究，逐步向大语言模型发展。本文第一部分依发展趋势系统总结了 NLP 领域技术方向分类，并在每个技术方向中给出了具体应用示例。由于笔者在组内实习的科研内容为开放域信息抽取任务(open information extraction, OIE)，故笔者希望可以在有限的篇幅中把一个 NLP 子领域分析清楚。故文章第二部分重点介绍 NLP 细分领域：（开放域）信息抽取。将介绍相关技术模型的发展历程、分类，并给出如今在 OIE 任务（在普遍使用的 benchmark 中）sota 模型 DetIE 的详细介绍。最后在文章小结笔者给出了自己对 NLP 如今发展的相关看法。

关键词:　自然语言处理;开放域信息抽取;信息抽取

1　NLP 现状及技术概览

自然语言处理技术发展日新月异。从技术角度来看，当前的语言模型已经进入了预训练（大）模型的时代，其在阅读理解、情感分析、信息抽取等各项自然语言处理任务上的表现已有巨大突破。目前自 NLP 相关前沿技术正在与图像、语音等其余基础 AI 技术进行结合，希望在更复杂、更契合实际生活多样情景的任务上取得良好效果。

从产业应用的角度来说，自然语言处理技术已经被应用于人机交互（HCI）、社交媒体、广告分析与投放等领域。随着技术与业界需求的发展，更多的新兴 AI 应用在逐渐衍生，NLP 也将随之在产业界拥有更多不同的地位和作用。未来 NLP 将与更多领域基于不同颗粒度的结合，为各行业创造更大的价值。

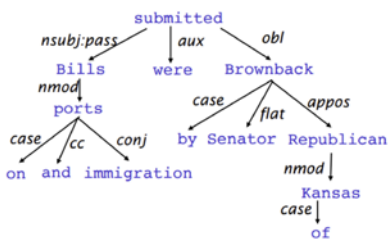
为了便于介绍，笔者把 NLP 领域技术方向划分为三个部分：自然语言理解（Natural Language Understanding，NLU）、自然语言生成（Natural Language Generation，NLG）及如今风靡的大语言模型（Large Language Model, LMM） [1] [2] 。以下将分别从三种任务类别展开，给出相关任务核心思想介绍并给出具体示例。

1.1 自然语言理解

NLU 是 NLP 最为基础的任务，也是 NLG 所必需的上游任务。NLU 旨在理解和分析人类语言，重点关注对文本数据的理解，通过对其进行处理来提取相关信息。同时 NLU 涵盖了 AI 面对的最困难的挑战：理解对话，进而提供直接的人机交互，并执行和语言理解相关的任务。从不同角度来“理解对话”，可以把 NLU 分为三个子任务。

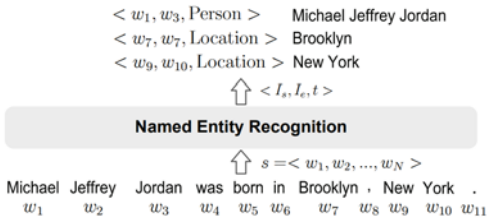
（1）自然语言认知与推理

自然语言认知与推理可以认为在语言学和语义层面识别、建模自然语言，并逐渐构建语义的过程。如词汇的语言学、形态学相关（形态学分割）、分块标注类（分词）、语句解析（依存句法分析）、话语分析等（识别句子 token 之间的关系）。下图给出极为经典 NLP 中依存句法分析的实例。



（2）文本挖掘与信息抽取

文本挖掘与信息抽取希望从深度语义（结合更大范围的上下文、人类知识常识等现实世界语义）来建模自然语言。文本挖掘希望从文档级特征抽取与快速检索（一般业界为了追求文本挖掘的效率，不把信息抽取作为其上游任务），信息抽取希望从语言中抽取结构化内容并最终提炼知识构建知识库（或本体 Ontology，但学界和工业界目前未给出可行方法）。下图为信息抽取任务中经典的命名实体识别（NER）任务的示例图。



（3）情感分析与文本风格分析

情感分析的基本任务是文本极性分类，即给定一段文本判断它所表达的情感是积极的、消极的，还是中性的。可以进一步细化为文本情感强度分类（或观点检测），即给定一段文本，判断它所表达的情感强度是强烈的、中等的还是弱的。通常情感分析采用基于规则的方法（预先定义词典或规则，基于文本中情感词、修饰词来判断情感倾向）或基于机器学习的方法（使用统计模型或 NN，从大量标注情感倾向和强度的文本数据中学习特征和规律）。基于此可以很自然拓展应用到文本风格分析之中。[3][4]

1.2 自然语言生成

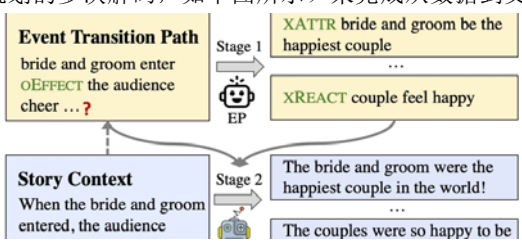
NLG 是生成包含意义和结构的短语、句子和段落的过程，通常是在机器已经完成了自然语言理解之后的下游任务。为了利用数据文本来生成一种人类可读格式的自然语言，NLG 通常由以下 3 个部分构成。**生成器**：负责根据给定的意图，选择与上下文相关的文本；**表示组件和层级**：为生成的文本赋予结构；**应用**：从对话中保存相关数据，从而遵循逻辑。根据应用场景的不同，概念自顶向下可把 NLG 划分成三个子领域。

（1）文本生成与写作

文本生成过程也即给定需求后自动补齐上下文。现在主流文本生成被看作是分类任务，也即每次词的生成可以看成在数千规模的词表（类别标签）上分类，当分类到停止符时就停止生成。形式上，就是生成式模型对联合概率密度的建模：

$$P(y) = P(y_1 y_2 y_3 \dots y_n) = P(y_1)P(y_2|y_1)P(y_3|y_2 y_1)\dots P(y_n|y_1 y_2 \dots y_{n-1})$$

常用的模型有词袋（BOW，忽略词间顺序）、自回归（用已生成的所有字符来生成新的字符）、非自回归（如基于规划的多次解码，如下图所示）来完成从数据到文本及文本到文本的生成

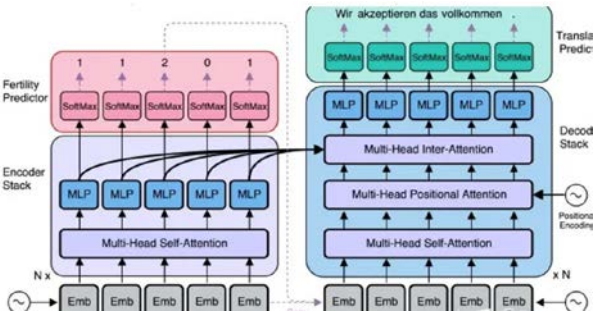


（2）机器翻译

机器翻译可以视作文本生成与写作中 sequence-to-sequence 的具体应用，由于其过于重要且有许多经典算法，故在此处单独列出。

机器翻译任务希望输入和输出文本之间具有相同的语义，但属于不同的语言。根据翻译方法可以分为以下几种：

- 概率统计机器翻译、神经网络机器翻译
- 多语言机器翻译（通过 encoder、decoder 把一种源语言翻译成多种目标语言）
- 并行解码和非自回归的机器翻译（并行解码是非自回归模型，无法不断考察之前时间节点间输出的关系再作为新的输入，而是一次性生成所有无法确定对应关系，优点是容易重复、漏翻。但生成结果的流畅度低等）。下图给出具体示例，通过 multi-head attention 机制实现并行解码。



（3）人机对话与问答

问答任务期待能够获得一个问题的精准解答，经常需要结合信息抽取、语义解析、知识库上的操作，故很多时候看的是 accuracy，F1-score 等精确性的指标。

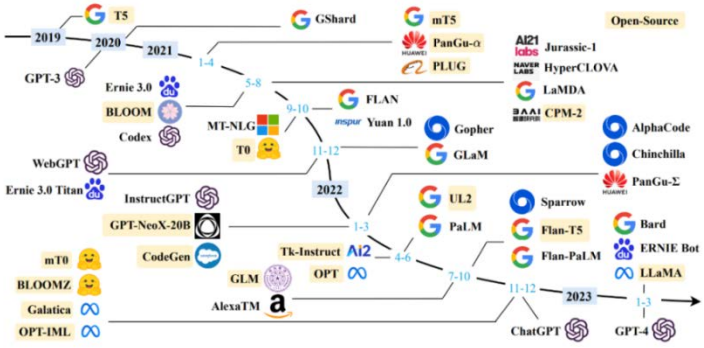
对话任务一部分任务像是一种“闲聊”，另一部分关注于完成一定目的（任务型对话、对话推荐）等。其更多关注如何生成让人们更感兴趣的内容，让对话更好地进行下去，并完成任务。故很多时候会看对话的轮数，人工评价等作为其指标。以下给出任务型对话的一个示例，其中使用了信息抽取任务结果并给出相关评价指标。

1.3 大语言模型

大语言模型（LLM）是基于海量文本数据训练的深度学习模型。它不仅能够生成自然语言文本，还能够深入理解文本含义，处理各种自然语言任务，如文本摘要、问答、翻译等。

同时 LLM 作为一种通用的机器学习方法在具体的任务上实践时，必须对相关参数进行微调以适应特定的使用环境。所以“预训练+微调”（pre-train + fine-tune）作为一种机器学习的新范式这些年来被广泛视作一种机器学习新范式。[5]

由于 LLM 更是一个庞大话题，故此处就不对具体模型做出过多解释。下图给出自 2019 年来超百亿参数量的大语言模型发展流程图，相关细节可以参考附录 1，表中记录自 Bert 和 Transformer 在 2018-2019 年提出后的大模型。[6]



2 （开放域）信息抽取任务

2.1 研究问题

信息抽取（information extraction，IE）任务见名知意，希望从自然语言表示的文本中抽取出结构化信息，并把其转化为关系元组：一个由参数集合和短语组成的元组，其表明了参数间的语义关系。具体形式如下图。

$$< arg_{1-1}, rel_1, arg_{1-2}; arg_{2-1}, rel_2, arg_{2-2}; \ldots; arg_{n-1}, rel_n, arg_{n-2} >$$

上图呈现出通用的抽取结果，但在进行对应 benchmark 评测时以及不同模型实际生成结果时未必如此呈现。由于抽取出二元关系+多元关系是完备的（completeness），即多个二元关系、多元关系组可以表达相同含义的多元元组。故在多个 IE 模型抽取结果中，抽取出<arg1, rel, arg2>的二元关系组和<arg1, arg2,…,argn,rel>的多元关系组集合也符合人们期待。[7]

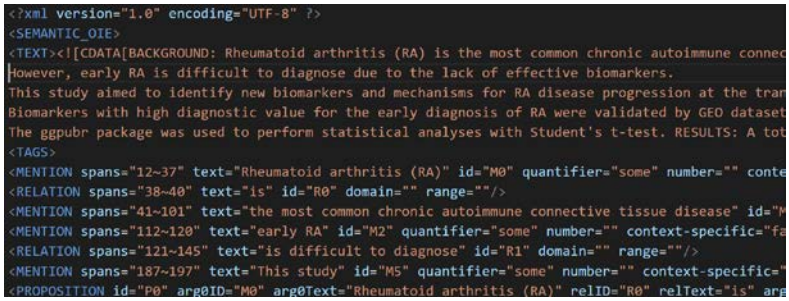
值得注意的是，传统 IE 只关注手工良好定义、狭窄的抽取模式（pattern），并且训练好的模型只能在小型、同质（相同领域）的语料库抽取，在大数据、任务多样化时代实际应用价值有限。基于此，研究人员提出新的抽取范式：**开放域信息抽取**（Open IE, **OIE**），OIE 首先从文本中提取出抽取模式，此后再利用它来提取关系元组。同时 OIE 任务不局限于发现定义好的小规模关系，而是要在无预先定义好抽取模式的文本中，提取发现的所有关系。主要会面临以下三大挑战：

(1) **自动化。**OIE 任务必须遵循无监督的策略，也即不可以提前定义好目标关系，结果元组中可能出现的关系必须从语料库中获取。

(2) 无法利用语法（句法）分析器辅助抽取。由于语料库异质性（Corpus Heterogeneity），只关心局部信息且只在训练语料及特定领域中才可良好工作的依存句法分析器无法作为抽取的辅助工具

(3) 效率。由于 OIE 需要对大量领域无关文本进行分析，且需从中获得抽取模式耗时很长。且部分抽取任务应用在实时系统中，所以 OIE 系统需要高效计算。（一种可行的思路是优先提取浅层语言特征，如使用 POS tag）

由于抽取符合需求、符合特定联系的关键字实体是各种 NLP 任务的基础，在如今许多（大）语言模型实际训练和测试对应的 pipeline 中，信息抽取都是必不可少的上游任务，也是后续进行如情感分析、自然语言语义分析、文本生成的前提。此外，我们组内（KR）也正在使用 OIE 技术来助力 OWL 本体（Ontology）自动构建，希望 NLP 可以改善 OWL 本体需要领域专家给出 query list 才能构建的现状。



本图为笔者使用 DetIE 模型（后面会介绍）从医学期刊摘要中的抽取结果（xml 格式）

2.2 OIE模型

2.2.1 非神经 OIE (Non-Nerual OIE)

在深度学习之前，传统的 OpenIE 系统通常是基于统计学的或基于规则的，并且严重依赖于语法模式的分析。以下介绍三种传统的 OIE 系统，并给出它们的代表模型。

（1）Textrunner (Learning-based Systems)

基于学习的 Textunner 于 2007 年提出，采用基于对比(Contrastive based)的自监督学习方式构建模型：学习对两个事物的相似或不相似进行编码来构建表征，即通过构建正样本（positive）和负样本（negative）后引入机器学习模型进行训练。Textrunner 采用相同思想，通过三个模块实现开放域信息抽取功能。

模块 1：识别并学习可信关系。给定少部分的文段，依据语法解析器 parser 来启发式把训练样本识别和标注成正、负训练样本，并将其作为输入给朴素贝叶斯分类器训练。此分类器使用非词汇化的 pos 标注+名词短语来学习可信关系。

模块 2：抽取器。识别名词短语，产生名词短语对。并把抽取出来的名词短语对给贝叶斯分类器，只保留被分类器认为是可信关系的短语。

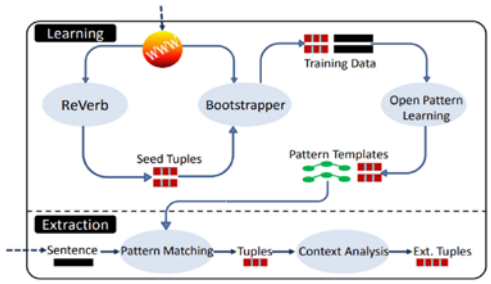
模块 3：冗余度分配概率。基于冗余度的评估器给保留的三元组分配概率。更好利用文本中的冗余度，出现多次的置信度更高。最终设置置信度保留组合出来的三元组。[8]

由于其模型极为经典且性能有限，常常作为新模型的 baseline 用作模型评估。

（2）REVERB (Rule-based Systems)

基于规则的 REVERB 于 2011 年提出，是使用了认为定义的抽取规则的一种浅提取器。REVERB 引入了句法约束（基于 POS 的正则表达式），其可以涵盖英文动词关系短语的 85%，进而减少了不连贯、无信息的提取；同时引入了词汇约束，认为合法的关系短语应该在大型语料库之中包含很多的相关论述，进而减少了过于细致、冗余信息的提取。[9]

基于 REVERB，此后又演化出 OLLIE（Learning-based Systems）。在 REVERB 的基础上获得更高精度的目标三元组集，并最终使用 wikipedia 作为训练数据源 bootstrap 了更大的训练集。具体细节不在这里展开，只给出模型的 pipeline 如下图所示。[10]



（3）ClausIE (Clause-based Systems)

基于从句的 ClausIE 系统于 2013 年提出，基于句子的重构把复杂句子（关系短语分布在几个子句上）转化成一组容易划分且语法简单的独立子句。ClausIE 系统利用了英语语言学知识，把输入句子的依赖关系映射到从句。其中每个从句的类型是通过把动词性质的知识（领域无关）和输入从句的结构结合来确定的。最终每个子句生成一个命题，每个命题是不同的信息片段。具体示例如下图所示。[11]

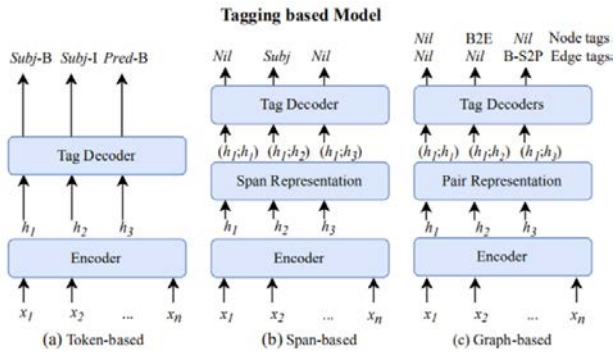
	Pattern	Clause type	Example	Derived clauses
S_1:	SV_i	SV	AE died.	(AE, died)
S_2:	SV_eA	SVA	AE remained in Princeton.	(AE, remained, in Princeton)
S_3:	SV_cC	SVC	AE is smart.	(AE, is, smart)
S_4:	$SV_{mt}O$	SVO	AE has won the Nobel Prize.	(AE, has won, the Nobel Prize)
S_5:	$SV_{dt}O_iO$	SVOO	RSAS gave AE the Nobel Prize.	(RSAS, gave, AE, the Nobel Prize)
S_6:	$SV_{ct}OA$	SVOA	The doorman showed AE to his office.	(The doorman, showed, AE, to his office)
S_7:	$SV_{ct}OC$	SVOC	AE declared the meeting open.	(AE, declared, the meeting, open)

2.2.2 神经 OIE (Nerual OIE)

随着深度学习技术的兴起，信息抽取领域也出现了更多的可能：许多 Neural Open IE 系统（利用多层神经网络助力 OIE）已被提出，并已实现相当大的性能提升。以下将把神经 OIE 系统分为两类，并分别介绍其下代表模型并对比性能。

2.2.2.1 基于标记的模型（Tagging -based Models，判别式模型）

基于标记的模型把 OIE 看成一个序列标记任务，也即模型给句子中每一个单词或一组单词贴标签（通常可以为 subject、predicate、argument 三种），模型学习每个单词的标签或基于句子的跨度对每个词的条件概率分布建模。一般此类模型包含三个模块：嵌入层（为单词生成向量）、编码器（将上下文信息融入单词表示）、标签解码器（基于单词标记及对应标记方法预测单词标签）。依据标记方案的不同，主要有三种实现思路：**基于单词的模型（下图 a）**，预测一个单词属不属于 argument 或 predicate；**基于单词跨度（一组相邻单词）的模型（下图 b）**，预测单词跨度是否是主语，谓词或参数；**基于图的模型（下图 c）**，构建一个 Graph 来识别三元组，其中节点为单词跨度，以及表示属于相同事实的连接节点的边，通过挖掘图中的极大团来提取元组，可以看做是是一种马尔科夫随机场。[12]



以下介绍基于 Tagging-baed 思想的 sota 模型 DetIE。2022 年被提出的 DetIE 源自对上一代 OIE 任务 sota 模型 OpenIE6 的优化，其受计算机视觉中基于单阶段锚点对象检测模型的启发，使得模型可以一次提取多个关系元组，并从不同的角度看待 OIE 任务，将其视为一个直接集预测问题。

$$input: \{x_1, x_2, \ldots, x_T\}$$
$$output: \{\{L_{1,1}, \ldots, L_{T,1}\}, \ldots, \{L_{1,N}, \ldots, L_{T,N}\}\}$$

思想上，DetIE 把抽取任务看成序列标注任务，输入是 T 个 tokens，对这每一个 token 进行 N 次标注，本次不需要预测的 token 作为掩码（mask），标注 C={ Background}，意为本次没有被覆盖掉的已知内容。需要预测的 token 标注从 C={ S, R, O}中选择。

整体上，模型使用了基于二分匹配的顺序不可知损失来强制进行唯一预测，并使用 Transformer 中的 encoder（也有无序性）进行序列标记。以下将对其进行介绍。

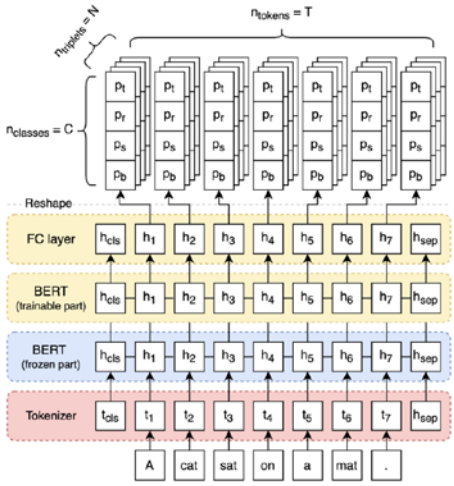
（1）模型架构：Bert + Transformer

模型架构基于 Bert 设计，对 Bert 进行 fine-tune 后从单个文本片段中提取大量预定义的概率掩码，每个掩码对应一个可能的三元组（training part）。此后 Bert 把每个 token 对应的 embedding 通过一个全连接神经网络，重新产生一个 T*N*C 的特征矩阵。

下图中存在一些 frozen part，经过研究人员实验发现把 Bert 各层级全部 frozen，并且在顶层加入一个额外的 Transformer 来进行序列标注性能更好。

模型输入：原始序列中字符。

模型输出是一个三维矩阵，呈形(T,N,C)。其中 T 为输入序列中 tokens 的数量，N 为提前定义好的需要提取三元组的个数（也是 N 次预测，每次预测只得到一个可能的三元组），C 为上述四类类别标记。其中 N 是提取三元组数量的上界，应保证足够大可以覆盖文档中所有可能的三元组，但数量不可过大否则会导致类别失衡使 Background 特别多，DetIE 综合多次试验选择 N=20。每一个三维坐标(t,n,c)确定唯一位置，该位置对应值为 DetIE 预测在第 n 次预测中第 t 个 token 属于类别 c 的概率。本次预测最终实际输出是从 4 个类别中选择概率最大作为其所属类别。



（2）损失函数：基于交叉熵的顺序不可知损失函数

基于 OIE 模型预训练时对应的损失函数一般为在每个预测掩码和最近的 ground truth 上先进行双射匹配，后计算交叉熵。形式上，设 N 是预先定义进行掩码的次数，M 为真正的三元组数量。损

失函数计算时从 N 个预测三元组中依置信度选出 M 个最相关预测，和 M 个预先人为设定好的 ground truth 进行双射匹配后做交叉熵损失，并希望最小化这个值。

但在对 OIE 模型进行预训练时，需要解决三元组的预定义排序问题（N 的个数，以及抽取出三元组对应排序），且精确计算容易产生阈值化（thresholding）。故研究人员设计了一个特殊的损失函数：基于交叉熵的顺序不可知损失函数，此后不精确计算模型对应损失而采用更为平滑的 IoU(Intersection over Union)。具体示例如下。

Input sequence
Life is a tale told by an idiot , full of sound and fury .

Predictions
1: (Life, is, a tale told by an idiot)
2: (a tale told by an idiot, is, full of sound and fury)
3: (an idiot, is, full of sound and fury)

Ground truth
1: (Life, is, a tale told by an idiot)
2: (a tale told by an idiot, is, full of sound and fury)

	gt_1	gt_2
$pred_1$	1.	0.05
$pred_2$	0.05	1.
$pred_3$	0.07	0.67

IoU 的概念来自于计算机视觉中对象类别分割问题的标准性能度量，在 OIE 问题中定义如下：模型计算得到的 N 个预测掩码对应（概率向量）和 M 个 ground truth（在此位置若提取则为 1）计算 IoU，求出对应结果矩阵大小 N*M。并最终使用匈牙利算法（Hungarian）最大化 IoU 总和并最终采用 Adam 随机梯度下降优化参数。具体公式及含义如下图所示。[13]

$$IoU_{n,m} = \frac{I_{nm}}{U_{nm}}$$

$$I_{nm} = \sum_{t,c} p_{tnc} \cdot l_{tmc} \quad （n：第\ n\ 次预测、\ m：第\ m\ 个\ groundtruth）$$

$$U_{nm} = \sum_{t,c} p_{tnc} + \sum_{t,c} l_{tmc} - I_{nm}$$

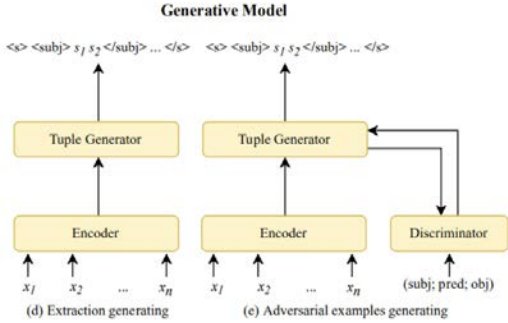
$I_{n,m}$：把所有真实 类别的预测概率全部加起来了

$U_{n,m}$：其中第 $1,3$ 项，即错误概率概率的加和；第 2 项即 N（$tokens$ 数量）

最终 DetIE 在 benchmark (IMoJIE、LSOIE、Synth)数据集上训练，并在数据集 (LSOIE、CaRB、MultiOIE2016)上进行测试，在 CaRB 数据集测试中 AUC 值最高，且在和以前 sota 模型 F1、AUC 值相差不大的情况下，在三个数据集上预测速度最快！

2.2.2.2 基于生成式的模型（Generative Modles，生成式模型）

基于生成式想法的模型将 OIE 任务表示为一个序列生成问题，它读取一个句子并输出一个序列的提取。可以认为模型是对一个单词序列 S 的联合分布建模，此后依此联合分布最大化条件（后验）概率 P(Y|S)，其中 Y 是期望提取的标注序列。



类似的，生成式模型也可以分为两种实现方法：**提取信息生成**和**对抗样本生成**。其中**提取信息生成**通常由 encoder 和 decoder 组成，encoder 给出句子上下文的向量表示，decoder 基于句子上下文和目前已生成序列

对 embedding 进行解码。可以和 AE、VAE、CVAE 在 OIE 任务中的应用；**对抗样本生成**基于对抗神经网络（GAN），同时包含 encoder、decoder 和 discriminator，具体内容在深度学习课程中已详细阐述，此处就不再赘述。

IMoJIE 是生成式模型的代表，也是目前基于生成式模型的 sota。由于模型提出时间较 DetIE 早且性能稍有欠缺，此处就不展开介绍，具体内容可参考文献。[14]

2.3 模型评估

OIE 任务中全面、客观、可复制进行模型评估是信息抽取技术的一大难点，其一是因为大部分方法使用专有数据集，甚至都不是领域依赖的小型语料库；此外大多数 OIE 系统都专注于英语而忽略了其他语言。故目前 OIE 系统都只在几百个小规模语料库上做手动评估，且很大程度上仅限于新闻、维基百科、Web 领域，对于各种文本类型 OIE 系统性能如何目前还未可知。

2.3.1 数据集和语料库

在现有论文中使用最多的是 OIE16、CaRB、Wire57-C、LSOIE 数据集系列，这些数据集是近些年针对 OIE 系统重新抓取并处理后的，有对应的抽取原文和 ground truth 且在多数模型上抽取质量较高。具体细节请见附录 2。

此外，OIE 系统需要额外配备语料库，方便在真实世界进行测试，常见语料库如 OPIEC、ReVerb extractions、PATTY、WiseNet (1.0 and 2.0)、KB-Unify，具体细节请见附录 3。

2.3.2 评价指标

早年基于统计的 OIE 系统通常只使用精度（precision）、F1 作为性能评价指标，随着深度学习的发展，AUC、正确抽取的数量、覆盖率作为评价指标也变得可行了，具体定义如下图所示。

$$\text{precision}_{\text{sys}} = \frac{\sum_{i}^{n}\left(\sum_k |t_i^{p_k} \cap g_{m(i)}^{p_k}|\right)}{\sum_i^n |t_i|}$$

$$\text{recall}_{\text{sys}} = \frac{\sum_{j}^{N}\left(\sum_k |t_{m(j)}^{p_k} \cap g_j^{p_k}|\right)}{\sum_j^N |g_j|}$$

$$F_{1\text{sys}} = \frac{2\, p_{\text{sys}}\, r_{\text{sys}}}{p_{\text{sys}} + r_{\text{sys}}}.$$

其中 g 表示 ground truth 抽取元组，t 表示系统真正抽取出来的元组结果，此时系统已经完成了从 ground truth 元组和系统抽取出来的元组间的映射，具体详细形式如下图所示用来定义准确率和召回率。此外 p 是系统的准确率，r 为系统的召回率，基于 p、r 可以用来定义 F1 值 [15] 。

$$t = (t^{a1}; t^r; t^{a2}; t^{a3}; \dots) = t^{p_k}_{p\in[1,n]}$$

其中 p_2 是 n 元联系中的联系，其余为实体

$$|ti| = |t_i^{a1}| + |t_i^r| + |t_i^{a2}| + |t_i^{a3}| + \cdots = \sum_k |t_i^{pk}|$$

2.4 小结

随着深度学习的发展，OIE 系统也日渐强大，不论在准确率、召回率、F1 值，还是抽取速度上相比传统基于统计的 IE 性能都有显著提升。但随深度学习的引入，OIE 系统的不确定性在提升，模型参数量也在逐

量级增加。基于上述 OIE 从非神经模型到神经模型的演化过程，以及各个模型不同的实践侧重点，我归纳出以下几点未来（开放域）信息抽取可能的发展方向，不一定正确望兼听则明。

（1）**更开放：**

- 当前的研究主要关注于在语句级别提取信息，未来可以在多种级别如文档级进行信息提取。
- 当前的研究主要关注于英语语料库，未来可以在多种语言上提取信息。（DetIE 已经在尝试，但仍是少数）
- 未来可以支持从半结构化或多模态数据中提取信息，扩展 IE 的提取能力。

（2） **更专注：**

- 传统 OIE 从源文本中提取所有的事实内容。然而在许多场景中，我们只对与某些主题/实体相关的事实感兴趣。因此更专注抽取某些实体/关系能够使得 OIE 提取的信息更好地为下游任务所使用。

（3）**更统一：**

- 希望能将 OIE 与 IE 很好的融合，使用 OIE 进行 IE 的预训练，利用其开放性和通用性来帮助 IE 模型更好地理解实体、关系等内容，利用 IE 的轻量化和可解释性来帮助 OIE 缩减模型规模和提高深度学习的通病“不可解释性”。

3 结语

自大语言模型诞生后，传统自然语言处理相关技术研究前景被部分研究人员不断否定，他们认为只需要 LLM 就可以代替所有传统 NLP 的研究方向。笔者在一定程度上认可相关观点，但值得注意的是传统 NLP 和大语言模型实际代表了人工智能的两种方向：通用智能和专业智能。虽然像 ChatGPT 这样的通用型 LLM 表现不俗，但是对于 NLP 的一系列问题，永远依赖于调用一个外部模型注定不可行。从机器学习中“没有免费的午餐”定理可知，对于特定的问题一定存在对应最优算法。而由于神经网络“万有逼近性”对应局限性，LLM 也不可能成为任意任务的最优解。此外，现阶段 LLM 的微调、生成训练数据依然需要传统 NLP 的相关知识和技术。可以预见，在相当长一段时间内像传统 NLP 那样细分垂直的子任务会越来越少，而多模态/跨模态的场景和应用会随之增多。

附录 1：继 Bert 和 Transformer 后大语言模型

公司\机构	模型	发表时间	解决问题	方法	结构	链接地址	模型规模(最大参数量)
GOOGLE	T5	2019	给预训练模型提供通用框架，把所有任务都转成一种形式	提出了 Encoder+Decoder 新结构，并结合 BERT+GPT 优势	Encoder+Decoder	https://jmlr.org/papers/v21/20-074.html	3B/11B
	LaMDA	2022	提升模型安全性、事实性，同时利用外部知识来源	众包，选择人类偏好的回答，利用标注数据来 fine-tune 模型	Decoder 结构	http://arxiv.org/abs/2201.08239	137B
	GLaM	2022	针对节约计算资源，推进细分专家领域的发展	混合专家（MoE）模型，每个模型针对不同的输入	MoE，Decoder 结构	http://arxiv.org/abs/2112.06905	1200B
	PaLM-E	2023	多模态接入视频传感器，把 LLM 应用到机器人领域	图像嵌入到语言标记相同的隐空间，基于 Transformer self-attention 进行处理	Encoder+Decoder	http://arxiv.org/abs/2303.03378	562B
	OPT-175B	2022	解决超大规模语言模型（千亿级别）开放性、访问便捷性问题	完全分片数据并行(FSDP) API 和 NVIDIA 的张量并行抽象在 Megetron-LM，来减少参数量	Decoder	https://arxiv.org/pdf/2205.01068.pdf	175B（简化后）
META	LLaMA	2023	使用更多 token 训练，更少的模型参数。（可运行在单 GPU 环境）	致力于于找到合适的数据量和参数量，以实现快速推理	Decoder	https://arxiv.org/abs/2302.13971	65B
OPENAI	GPT-GPT3.5	2020	解决 gpt2 使用 zero-shot 训练不佳的问题，对 dense attention 优化	Few-shot + sparse attention	传统 GPT 结构+引入 sparse attention 模块	https://proceedings.neurips.cc/paper_files/paper/2020/file/1457c0d6bfcb4967418bfb8ac142f64a-Paper.pdf	1750B
	GPT4	2023	解决大规模的多模态模型。提升了利用知识去解决具体问题能力	使用人类反馈强化学习 (RLHF) 来 fine-tune	同 GPT-3	https://arxiv.org/abs/2303.08774	/
清华	ChatGLM	2022	结合了 GPT 和 BERT 类模型的优点，并将 NLU 任务转换成生成任务	使用中英双语语料训练，在稳定性和性能方面进行了调优。在模型结构上结合了 GPT 和 BERT。	基于自回归的空白填充，随机删除连续的 token(自编码)，并训练模型以顺序重建删除的 token（自回归）	https://aclanthology.org/2022.acl-long.26	130B

附录 2： OIE 任务各模型对比

	模型	发表时间	所属类别	评价指标	抽取速度 (SENT./SEC.)
非神经 OIE	TEXTRUNNER	2007	Learning-based	正确率	/
	REVERB	2011	Rule-based	准确率、召回率 R	/
	OLLIE	2012	Clause-baed	AUC	8.5
	ClauIE	2013	Clause-baed	准确率、AUC	4.0
神经 OIE	IMoJIE	2020	Tagging-based	F1、AUC	2.6
	DetIE	2022	Generative	F1、AUC	708.6

附录 3：OIE 任务训练、验证、测试数据集

名称	数量(单位：TRIPLES)	文本来源	抽取形式	下载链接
OIE16*	3,180	Wiki, Newswire	(arg1,rel,arg2)	https://github.com/jzbjyb/oie_rank
CARB*	3180 (annotate 1,282 sentences in dev and test by Amazon Mechanical Turk)	Wiki, Newswire	(arg1,rel,arg2)	https://github.com/dair-iitd/CaRB/tree/master/data
WIRE57-C*(2019)	34704	Wikipedia articles	(arg1,rel,arg2)	https://paperswithcode.com/dataset/wire57
LSOIE*(2021)	47,998	Science	(arg1,rel,arg2)	https://github.com/Jacobsolawetz/large-scale-oie/tree/master/data

附录 4：OIE 任务常见语料库

名称	数量(单位：TRIPLES)	文本来源	抽取形式	下载链接
OPIEC	341M	English Wikipedia	(subj, obj, rel)	https://www.uni-mannheim.de/dws/research/resources/opiec/
REVERB EXTRACTIONS	15 M	ClueWeb09 corpus	(subj, rel, obj)	http://reverb.cs.washington.edu/reverb_clueweb_tuples-1.1.txt.gz
PATTY	15M	WikiPedia articles	(subj, rel, obj)	https://www.mpi-inf.mpg.de/departments/databases-and-information-systems/research/yago-naga/patty/
WISENET (1.0 AND 2.0)	15M (annotated by humans)	WikiPedia articles	(subj, rel, obj)	http://lcl.uniroma1.it/wisenet/
KB-UNIFY	564MB	/	(subj, rel, obj)	http://lcl.uniroma1.it/kb-unify/

References:

[3] Begum Yilmaz, Sentiment Analysis Datasets, https://research.aimultiple.com/sentiment-analysis-dataset/，2024

[7] Niklaus C, Cetto M, Freitas A, et al. A survey on open information extraction[J]. arXiv preprint arXiv:1806.05599,2018.

[8] Yates A, Banko M, Broadhead M, et al. Textrunner: open information extraction on the web[C]//Proceedings of Human Language Technologies: The Annual Conference of the North American Chapter of the Association for Computational Linguistics (NAACL-HLT). 2007: 25-26.

[9] Fader A, Soderland S, Etzioni O. Identifying relations for open information extraction[C]//Proceedings of the 2011 conference on empirical methods in natural language processing. 2011: 1535-1545.

[10] Mausam, Michael Schmitz, Stephen Soderland, Robert Bart, and Oren Etzioni. 2012. Open language learning for information extraction. In Proceedings of the 2012 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning, pages 523–534, Jeju Island, Korea, July. Association for Computational Linguistics.

[11] Luciano Del Corro and Rainer Gemulla. 2013. Clausie: Clause-based open information extraction. In Proceedings of the 22Nd International Conference on World Wide Web, pages 355–366, New York, NY, USA. ACM.

[12] Zhou S, Yu B, Sun A, et al. A survey on neural open information extraction: Current status and future directions[J]. arXiv preprint arXiv:2205.11725, 2022.

[13] Vasilkovsky M, Alekseev A, Malykh V, et al. Detie: Multilingual open information extraction inspired by object detection[C]//Proceedings of the AAAI Conference on Artificial Intelligence. 2022, 36(10): 11412-11420.

[14] Kolluru K, Aggarwal S, Rathore V, et al. Imojie: Iterative memory-based joint open information extraction[J]. arXiv preprint arXiv:2005.08178,2020.

[15] Léchelle W, Gotti F, Langlais P. Wire57: A fine-grained benchmark for open information extraction[J]. arXiv preprint arXiv:1809.08962, 2018.

附中文参考文献：

[1] 人邮异步社区，什么是自然语言处, https://blog.csdn.net/epubit17/article/details/118310361, 2021

[2] Erutan Lai，自然语言的主要研究方向, https://zhuanlan.zhihu.com/p/605412919, 2023

[4] 自然语言处理之情感分析, https://baijiahao.baidu.com/s?id=1763027141682167823&wfr=spider&for=pc, 2023

[5] 一文读懂“大语言模型, https://zhuanlan.zhihu.com/p/644183721, 2023

[6] 自然语言大模型介绍, https://zhuanlan.zhihu.com/p/618786499, 2023