---------------------------------------------问题分析------------------------------------------------------

问题：基于输入文本的依存句法树，抽象出具有代表性的语法路径模式，形成规则库，用于后续在 Prompt 中引导 LLM 准确识别三元组或关系结构

整体研究思路：利用依存句法分析自动识别句法结构，构建规则库以抽象常见实体关系模式，再将规则信息嵌入到Prompt中，指导大语言模型（如ChatGPT、DeepSeek等）在抽取过程中进行结构感知与语义对齐

==========理想规则库==========

主要生成实体与关系的约束规则：

* 关系规则约束（依存句法分析）：帮助LLM判断属于哪种关系
* 实体约束边界规则（词典/LLM语义理解/边界约束规则）：用来明确哪些词可作为实体被抽取

一条规则示例：

[

{

"relation\_type": "所属关系", // 映射到你们定义的目标关系之一

"dependency\_pattern": "nsubj→VERB→nmod:of", // 依存路径结构

"trigger\_words": ["属于", "隶属", "是...的"], // 典型触发词，用于判定关系

"entity\_1\_type": "组织", // 限定抽取的实体类型

"entity\_2\_type": "上级机构",

"example\_sentence": "该学院属于南京大学。",

"matched\_span\_template": ["实体1", "关系", "实体2"], // 用于 Prompt 构造的结构模板

"notes": "适用于表达从属关系的主谓介宾结构" // 可选说明

}

{

"id": "R-101",

"type": "boundary\_rule",

"entity\_type": "研究方法",

"prefix": ["一种基于", "新提出的"],

"suffix": ["方法", "策略"]

}

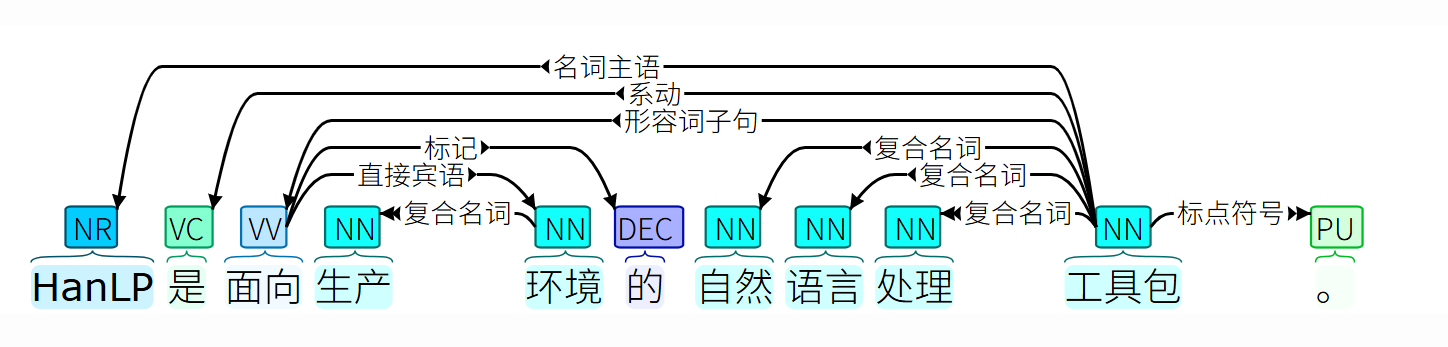
]

-----------------------------------------------依存句法分析原理---------------------------------------

定义：依存句法分析（Dependency Parsing、DEP）是一种分析一个句子中单词与单词之间的语法关系，并将其表示为树形结构的任务。

示例：

HanLP是面向生产环境的自然语言处理工具包。



-------------《基于依存句法规则的信息技术项目图谱构建》文献解读------------------

基于此文献获得实体的方法，可用来做实体边界约束

=====构建依存句法结构与实体扩展=====

* 目标：将单个词元扩展得到候选实体
* 工具选择：

使用依存句法分析工具（如 LTP 或 HanLP 等）对句子进行分析，得到每个词的：

词性（如 NN 名词、JJ 形容词）

依存关系（如 ATT 修饰、SBV 主谓等）

依存弧信息（词与其依赖词之间的语法关系）

* 构建方式：

将每个句子抽象成一棵依存句法树；

节点为词，边为依存关系；

每个名词被视为潜在实体中心，向上或向下遍历其依存连接词以扩展实体边界

* 扩展流程：

1. 以名词为核心的实体初始节点
2. 四类实体扩展规则

1）**词性规则**：

规则：中心词若被形容词、副词、数词等修饰，则将修饰词并入实体

例：一种创新的算法 → 实体为“一种创新的算法”

2）**依存关系规则**：

规则：若中心词与修饰词之间存在 ATT（定中关系）、ADV（状语）、SBV（主谓）、VOB（动宾）等依存关系，可扩展为连续片段

例：基于卷积神经网络的图像识别技术 → 保留 基于…的 + 核心名词，得到完整短语

3）**位置规则**：

规则：在依存树中，若修饰词在中心词左侧并紧邻，默认其构成实体前缀；

解决不连续短语问题

4）**特殊规则**（实体模板）：

识别出领域常见短语结构模板，如：

“基于XXX的YYY方法”

“XXX系统”

“XXX平台”

=======候选实体融合=======

* **目标**：候选实体进行融合重组，得到无交叉无重复的实体信息
* **线段排序融合算法**：合并多个规则结果

多条规则生成的实体可能交叉或重叠；

引入“线段排序融合”算法，根据实体起止位置排序并合并冲突部分，保留最优边界；

例如：

候选实体A：基于图像的

候选实体B：图像的识别技术

→ 合并为：基于图像的识别技术

-------------------------------------------------实验环节--------------------------------------------------

=====实体与关系类型定义与json格式存储=====

实体关系类型定义见 <实体定义.xlsx> <关系定义.xlsx>

将实体与关系以json格式存储

代码：entities.py、relations.py

结果：entities.json、relations.json

=====依存句法分析=====

注意：

1. 论文、作者、单位、发表时间仍以规则抽取模式进行抽取，不参与大模型抽取环节，所以不必写入规则库
2. 此处依存句法分析可以可视化（论文中或许有需要）



* 官方词表：

Stanford Dependencies Chinese — HanLP Documentation：斯坦福依存关系对照表

ctb — HanLP Documentation：词性标注对照表

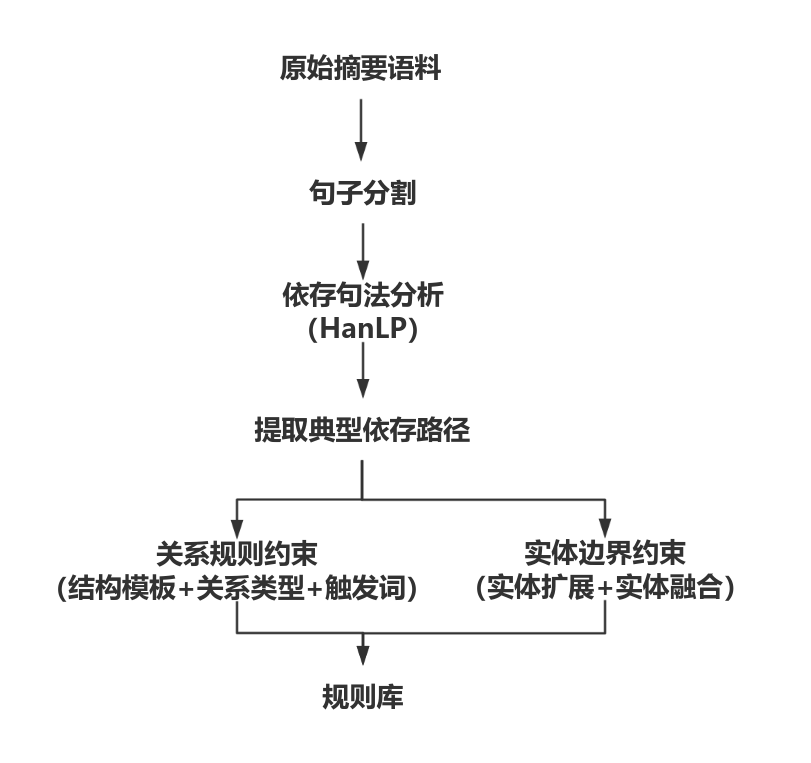
* 数据源的选择：以“摘要”为主 + 补充“含有目标实体”的段落

这里先以217篇PHM领域的摘要文本作为数据源

* 数据预处理：从题录信息抽取文章标题与摘要文本存储为json格式

代码：extract\_summary.py

结果：summaries.json



文献查找：提取典型依存路径

* 依存句法分析

对每一篇文献的摘要：

1. 分句处理：首先以。！？作为划分标准，接着以；或，再次划分
2. 分词处理：对每一句进行分词处理
3. 对每一句做依存句法分析得到该句各个词语与其父节点的的依存关系
4. 每篇文献依存句法分析结果单独存储为“title”+“\_dependency”

代码：analyze\_dependencies.py

结果：dependency\_results文件夹下内容

结果说明：

-title：文章标题

-summary：该篇文献摘要

-analyzed\_sentences：包含每条子句的依存句法分析结果

-sentence：分句后的具体子句

-tokens：该子句分词后的词列表

-parsed：该子句依存句法分析结果

-id

-form：分词后的具体词

-head：该词语的父节点的id

-deprel：该词语与其父节点词语的依存关系

-pos：词性

* 提取典型依存路径

定义：依存路径是指，在依存句法树中，连接两个实体（词元）之间的路径关系（包含路径上的依存关系和词性）

研究方法 ——[nsubj]→ 提出 ——[dobj]→ 模型

可形成三元组抽取模式：(研究方法, 提出, 模型)

依存路径提取路线：

1. 确定实体对

从句子中找出所有可能的实体对组合（实体1，实体2）

1. 依存图中找“最短路径”

·在 HanLP 的依存分析结果中构建依存图（树）

·使用 BFS / DFS 寻找 实体1 到 实体2 的最短依存路径

·保留路径中每个节点的：词、词性、依存关系、方向

1. 路径格式化

·实体1 -[nsubj]→ 动词（谓词） -[dobj]→ 实体2

或[

{"word": "研究方法", "dep": "nsubj", "upos": "NN"},

{"word": "提出", "dep": "root", "upos": "VV"},

{"word": "模型", "dep": "dobj", "upos": "NN"}

]

1. 统计并归纳典型路径模式

·统计最频繁出现的依存路径结构

·提取结构上类似的抽象模式

如：方法 –[nsubj]→ 动词 –[dobj]→ 对象

【Q1】依存句法分析是以“句子”为单位做的，即构建出的依存句法树也是针对一个特定的子句，所以一篇文献的摘要可能被分成若干字句，就有若干依存句法树，如果实体对在一棵树上即在某一子句内，或许可以找到其依存路径，但是对于“跨句”的实体对，应该如何处理

【Q2】实体在全文依存图上对齐，对齐后算是多个叶子，如何确定中心

【A1】全文增强依存图：拼接id 、head实现重映射，实现树间连接\*

【A2】一般以头部ID或者尾部ID作为中心

=====关系约束规则=====

=====实体边界约束规则=====

四类简单规则：

| **类别** | **规则名称** | **内容描述** | **示例** |
| --- | --- | --- | --- |
| **① 词性规则** | 修饰性词性扩展 | 若实体中心词（一般为名词）被**形容词（JJ）**、**数词（CD）**、\*\*副词（AD）\*\*等修饰词修饰，应将修饰词并入实体短语中 | “一种创新的算法” → 实体为 **“一种创新的算法”** |
| **② 依存关系规则** | 依存弧结构扩展 | 若实体中心词与其他词之间存在 ATT（定中）、ADV（状语）、SBV（主谓）、VOB（动宾）等依存关系，可向这些方向扩展，合并为完整短语 | “基于CNN的图像识别技术” → 保留 **“基于CNN的图像识别技术”** |
| **③ 位置规则** | 邻接词扩展 | 若修饰词在中心词左侧且紧邻，默认构成实体短语的前缀；特别适用于无“的”连接的名词短语识别 | “深度神经网络模型” → 实体为 **“深度神经网络模型”** |
| **④ 模板规则** | 实体模板识别 | 针对领域常见短语结构（如“基于…的…方法”、“…平台”、“…系统”），使用正则或模板直接识别为整体实体 | “基于LSTM的故障诊断方法” → 实体为 **“基于LSTM的故障诊断方法”** |

规则组合方式：

| **组合类型** | **规则组合结构** | **示例** | **说明** | **优先级** |
| --- | --- | --- | --- | --- |
| R1. **模板规则优先匹配** | 模板结构（正则或短语模板） | 基于CNN的图像识别方法 | 一旦命中即保留，不再向下扩展，适用于领域高频实体 | 1 |
| R2. 模板**规则 + 依存链** | 形如“基于XX的YY”中形成连续链 | 基于深度学习的预测模型 | 可迭代递归连接 ATT/ADV/VOB 依存结构 | 2 |
| R3. **依存 + 词性规则组合** | ATT + 修饰词性（JJ/CD/NN） | 一种创新的算法 | ATT是结构骨架，词性控制扩展边界 | 3 |
| R4. **依存 + 位置规则组合** | ATT + 相邻修饰词 | 神经网络算法 | 确保是句法依附，位置紧邻保证流畅性 | 4 |
| R5. **词性 + 相邻位置组合**（低优先） | 连续修饰词（无依存） | 复杂多维系统 | 作为回退机制，补全“未命中依存”的情况 | 5 |

PHM领域常见模板规则（chatgpt专家归纳）：