依存关系和深层语义

— 依存树(Dependency Tree), 语义角色(Semantic Role), and 最短依存路径(the Shortest Dependency Path)

Jingbo Xia

Huazhong Agricultural University

信息学院：生物信息、人工智能专业

2026-11-05

Table of contents I

1. 1. 实体识别 3
   - 1.1 实体识别性能的评估 4
   - 1.2 提升实体识别与信息抽取的方向 12
2. 2. 依存关系 Dependency relation 14
   - 2.1 依存关系在语言学标注中的层级位置 15
   - 2.2 斯坦福依存树 / Stanford Dependency Tree 18
   - 2.3 一个例句 37
   - 2.4 进阶1：主谓宾关系抽取（SVO） 52
   - 2.5 进阶2：最短依存路径（SDP, Shortest Dependency Path） 57
3. 3. 语义角色与深层语义 61
   - 3.1 语义角色的定义 62
   - 3.2 医药文献知识挖掘：语义角色对深层语义的建模 67
   - 3.3 语料库和评测任务 73
4. 4. 代码实现 78
   - 4.1 最短依存路径——SpaCy for Dependency Tree & Networkx 83
   - 4.2 主谓宾关系抽取（SVO）——对应 2.4 进阶1 91
   - 4.3 从主谓宾出发：知识条目的标准化处理 104

Outline I

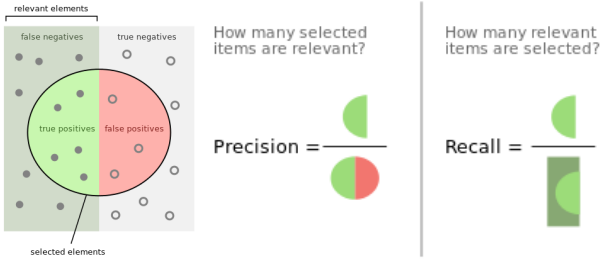
1. 1. 实体识别 3
   - 1.1 实体识别性能的评估 4
   - 1.2 提升实体识别与信息抽取的方向 12
2. 2. 依存关系 Dependency relation 14
   - 2.1 依存关系在语言学标注中的层级位置 15
   - 2.2 斯坦福依存树 / Stanford Dependency Tree 18
   - 2.3 一个例句 37
   - 2.4 进阶1：主谓宾关系抽取（SVO） 52
   - 2.5 进阶2：最短依存路径（SDP, Shortest Dependency Path） 57
3. 3. 语义角色与深层语义 61
   - 3.1 语义角色的定义 62
   - 3.2 医药文献知识挖掘：语义角色对深层语义的建模 67
   - 3.3 语料库和评测任务 73
4. 4. 代码实现 78
   - 4.1 最短依存路径——SpaCy for Dependency Tree & Networkx 83
   - 4.2 主谓宾关系抽取（SVO）——对应 2.4 进阶1 91
   - 4.3 从主谓宾出发：知识条目的标准化处理 104

Outline I

1. 1. 实体识别 3
   - 1.1 实体识别性能的评估 4
   - 1.2 提升实体识别与信息抽取的方向 12
2. 2. 依存关系 Dependency relation 14
3. 3. 语义角色与深层语义 61
4. 4. 代码实现 78

F1分数是信息检索任务中的常用评估指标，它由精确率（precision, p）和召回率（recall, r）的调和平均定义：

1. 精确率 Precision：正确识别出的正样本数 $\div$ 模型返回的全部正样本数 $=$ TP/ TP+FP



2. 召回率 Recall：正确识别出的正样本数 $\div$ 数据集中全部相关样本数 $=$ TP/ TP+FN ，
3. 两者的调和平均即为F1： $F_1 = \left(\frac{2}{\mathrm{recall}^{-1}+\mathrm{precision}^{-1}}\right) = 2 \cdot \frac{\mathrm{precision}\cdot\mathrm{recall}}{\mathrm{precision}+\mathrm{recall}}$.

课堂思考

这段话里一共出现了几个基因？几个性状？请你用不同颜色笔在纸上圈出来。

一篇模拟的水稻耐旱性研究摘要，作者的目标是**找出与耐旱相关的基因和性状**：

> Drought stress is a major abiotic factor limiting grain yield in rice (*Oryza sativa*); transgenic lines yielded stable performance under field conditions. We investigated the roles of OsDREB1A, OsDREB1B, and OsNAC1 in regulating drought tolerance. Transgenic lines overexpressing WRKY45 showed enhanced tillering under stress conditions, but no significant change in heading date. Quantitative RT-PCR revealed that these genes form a coordinated regulatory cascade responding to water deficit through the DRE/CRT element in the promoter region.

课堂思考

数一数，你圈对了吗？红色为基因，绿色加下划线为性状。

一篇模拟的水稻耐旱性研究摘要，作者的目标是**找出与耐旱相关的基因和性状**：

> Drought stress is a major abiotic factor limiting **grain yield** in rice (*Oryza sativa*); transgenic lines yielded stable performance under field conditions. We investigated the roles of **OsDREB1A**, **OsDREB1B**, and **OsNAC1** in regulating **drought tolerance**. Transgenic lines overexpressing **WRKY45** showed enhanced **tillering** under stress conditions, but no significant change in **heading date**. Quantitative RT-PCR revealed that these genes form a coordinated regulatory cascade responding to water deficit through the DRE/CRT element in the promoter region.

基因　性状

性状释义： **grain yield**（谷物产量）　**drought tolerance**（耐旱性）　**tillering**（分蘖）　**heading date**（抽穗期）

人工标注的 ground truth（8 个实体）

基因：`OsDREB1A`、`OsDREB1B`、`OsNAC1`、`WRKY45`　**性状**：`drought tolerance`、`grain yield`、`tillering`、`heading date`

问题

如果用两种不同方法做实体识别，输出如下，分别算 precision / recall / F1。

（PubTator只覆盖Gene/Disease/Chemical六类，**不识别性状**）

PubTator返回（4条基因）

✓OsDREB1A ✓OsDREB1B ✓OsNAC1 ✓WRKY45

性状全部漏检：drought tolerance、grain yield、tillering、heading date 一个也没抓到

TP = 4 FP = 0 FN = 4 TN：不适用（实体抽取没有"真负例"的定义）

它"吐出"的标注（可视化）：

Drought stress is a major abiotic factor limiting **grain yield** in rice (*Oryza sativa*); transgenic lines yielded stable performance under field conditions. We investigated the roles of **OsDREB1A**, **OsDREB1B**, and **OsNAC1** in regulating **drought tolerance**. Transgenic lines overexpressing **WRKY45** showed enhanced **tillering** under stress conditions, but no significant change in **heading date**. Quantitative RT-PCR revealed that these genes form a coordinated regulatory cascade responding to water deficit.

（只有4个基因被标红；所有性状——grain yield、drought tolerance、tillering、heading date——均未被识别。）

（基因用PubTator，性状用RiceMind词典——第8讲讨论过的路线A+B）

组合系统返回（7条，其中1条是误检）

✓OsDREB1A ✓OsDREB1B ✓OsNAC1 ✓WRKY45
✓drought tolerance ✓grain yield **X** yielded ✓tillering
TP = 7 FP = 1 FN = 1 TN：不适用（实体抽取没有"真负例"的定义）

它"吐出"的标注（可视化）：

Drought stress is a major abiotic factor limiting **grain yield** in rice (*Oryza sativa*); transgenic lines **yielded** stable performance under field conditions. We investigated the roles of **OsDREB1A**, **OsDREB1B**, and **OsNAC1** in regulating **drought tolerance**. Transgenic lines overexpressing **WRKY45** showed enhanced **tillering** under stress conditions, but no significant change in **heading date**. Quantitative RT-PCR revealed that these genes form a coordinated regulatory cascade responding to water deficit.

（漏检：heading date；误检：yielded——此处是动词"产出"，词典把 yield 的词形变化也匹配上了。）

公平比较：两个系统都在同一套金标准（8个实体）上评估

系统	TP	FP	FN	Precision	Recall	F_1
A：纯PubTator	4	0	4	$4/4 = 1.00$	$4/8 = 0.50$	0.67
B：PubTator+词典	7	1	1	$7/8 = 0.875$	$7/8 = 0.875$	0.875

课堂思考

系统B中漏掉的是`heading date`（词典没收录这个同义词），误检的是`yielded`（动词被当成名词"产量"）。怎么解决？

Outline I

1. 1. 实体识别 3
   - 1.1 实体识别性能的评估 4
   - 1.2 提升实体识别与信息抽取的方向 12
2. 2. 依存关系 Dependency relation 14
3. 3. 语义角色与深层语义 61
4. 4. 代码实现 78

实体识别的性能可以从哪些维度提升？

1. 更进一步的代码技能
   例如：正则表达式，基于规则的做法.
2. 机器学习方法，特征工程.
   例如：语言学特征.
3. 深度学习方法
   例如：嵌入表示.
4. 大语言模型（LLM）方法
   例如：Kimi, ChatGPT.

Outline I

1. 1. 实体识别 3
   - 1.1 实体识别性能的评估 4
   - 1.2 提升实体识别与信息抽取的方向 12
2. 2. 依存关系 Dependency relation 14
   - 2.1 依存关系在语言学标注中的层级位置 15
   - 2.2 斯坦福依存树 / Stanford Dependency Tree 18
   - 2.3 一个例句 37
   - 2.4 进阶1：主谓宾关系抽取（SVO） 52
   - 2.5 进阶2：最短依存路径（SDP, Shortest Dependency Path） 57
3. 3. 语义角色与深层语义 61
   - 3.1 语义角色的定义 62
   - 3.2 医药文献知识挖掘：语义角色对深层语义的建模 67
   - 3.3 语料库和评测任务 73
4. 4. 代码实现 78
   - 4.1 最短依存路径——SpaCy for Dependency Tree & Networkx 83
   - 4.2 主谓宾关系抽取（SVO）——对应 2.4 进阶1 91
   - 4.3 从主谓宾出发：知识条目的标准化处理 104

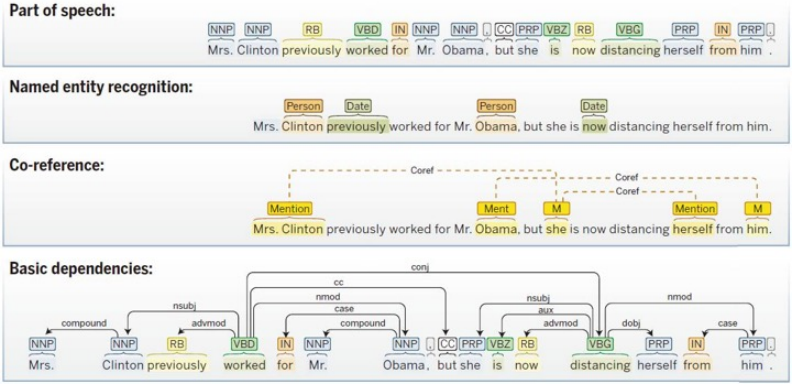
Outline I

1. 1. 实体识别 3

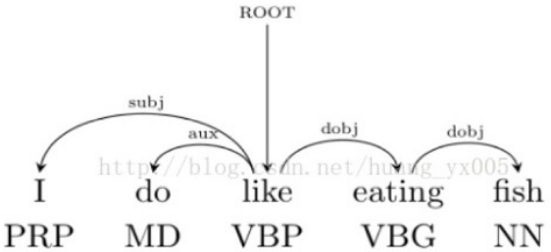
2. 2. 依存关系 Dependency relation 14
   - 2.1 依存关系在语言学标注中的层级位置 15
   - 2.2 斯坦福依存树 / Stanford Dependency Tree 18
   - 2.3 一个例句 37
   - 2.4 进阶1：主谓宾关系抽取（SVO） 52
   - 2.5 进阶2：最短依存路径（SDP, Shortest Dependency Path） 57

3. 3. 语义角色与深层语义 61

4. 4. 代码实现 78



说明：句子“Mrs. Clinton previously worked for Mr. Obama, but she is now distancing herself from him.”自上而下展示了递进的标注层级：
词性标注（Part Of Speech）、
命名实体识别（Person）、
共指消解（Coref：Mrs. Clinton—she—herself，Mr. Obama—him），
基本依存关系（basic dependencies，如 nsubj、dobj、nmod、conj 等）。依存关系位于标注链的最末端，刻画词与词之间的句法结构。



说明：句子“I do like eating fish”的依存树：
ROOT 指向谓语 like；
like 的 nsubj 为 I、aux 为 do、dobj 为 eating；
eating 的 dobj 为 fish。

词下方的 PRP、MD、VBP、VBG、NN 为词性标注；箭头由中心词（head）指向依存词（dependent），箭头上方标注关系类型。

Outline I

1. 1. 实体识别 3

2. 2. 依存关系 Dependency relation 14
   - 2.1 依存关系在语言学标注中的层级位置 15
   - 2.2 斯坦福依存树 / Stanford Dependency Tree 18
   - 2.3 一个例句 37
   - 2.4 进阶1：主谓宾关系抽取（SVO） 52
   - 2.5 进阶2：最短依存路径（SDP, Shortest Dependency Path） 57

3. 3. 语义角色与深层语义 61

4. 4. 代码实现 78

Stanford typed dependencies 表示法的设计目标，是用一种简单直观的方式描述句子中的语法关系，让非语言学背景的研究者也能轻松理解并有效使用，从而从文本中抽取关系信息[a].
该体系目前包含约50种语法关系。

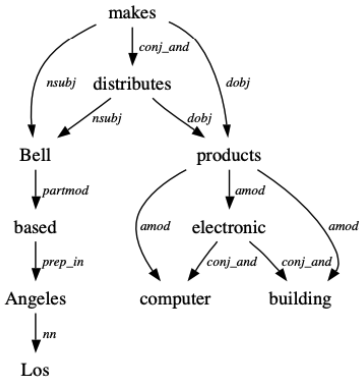
（手册原文：*The Stanford-typed dependencies representation was designed to provide a simple description of the grammatical relationships in a sentence that can easily be understood and effectively used by people without linguistic expertise who want to extract textual relations. The current representation contains approximately 50 grammatical relations.*）

[a] Stanford typed dependencies manual, https://nlp.stanford.edu/software/dependencies_manual.pdf.

斯坦福依存树：一个例子 我们来看一个英文句子：
"Bell, based in Los Angeles, makes and distributes electronic, computer and building products.".

句子："Bell, based in Los Angeles, makes and distributes electronic, computer and building products."

nsubj(makes-8, Bell-1)
nsubj(distributes-10, Bell-1)
vmod(Bell-1, based-3)
nn(Angeles-6, Los-5)
prep_in(based-3, Angeles-6)
root(ROOT-0, makes-8)
conj_and(makes-8, distributes-10)
amod(products-16, electronic-11)
conj_and(electronic-11, computer-13)
amod(products-16, computer-13)
conj_and(electronic-11, building-15)



句子："Bell, based in Los Angeles, makes and distributes electronic, computer and building products."

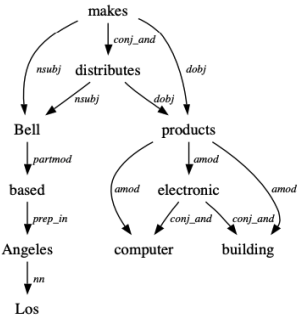
依存关系中的Head和Dep

head（支配词）：在依存关系中起主导作用的词；每个 token 都指向它的 head，形成有向依存边。

dep（依存关系标签）：标注该 token 与 head 之间的语法关系类型（如 nsubj、dobj、compound 等）。

右图每条标注形如 `dep(head-index, token-index)`：例如 `nsubj(makes-8, Bell-1)` 表示 `Bell` 的 head 是 `makes`，关系标签为 `nsubj`（名词主语）。

nsubj(makes-8, Bell-1)
nsubj(distributes-10, Bell-1)
vmod(Bell-1, based-3)
nn(Angeles-6, Los-5)
prep_in(based-3, Angeles-6)
root(ROOT-0, makes-8)
conj_and(makes-8, distributes-10)
amod(products-16, electronic-11)
conj_and(electronic-11, computer-13)
amod(products-16, computer-13)
conj_and(electronic-11, building-15)



nsubj: nominal subject/名词主语

A nominal subject is a noun phrase which is the syntactic subject of a clause. The governor of this relation might not always be a verb: when the verb is a copular verb, the root of the clause is the complement of the copular verb, which can be an adjective or noun. / 名词主语是充当从句句法主语的**名词短语**。该关系的支配词（governor）不一定总是动词：当动词为系动词时，从句的根节点是系动词的补语（complement），补语可以是形容词或名词。

例：*Clinton defeated Dole*（克林顿击败了多尔）——*Clinton* 是 *defeated* 的 nsubj；*The baby is cute*（婴儿很可爱）——*baby* 是 *cute* 的 nsubj（系动词 *is* 后，根节点为形容词补语 *cute*）。

“Clinton defeated Dole”	*nsubj*(defeated, Clinton)
“The baby is cute”	*nsubj*(cute, baby)

vmod: reduced non-finite verbal modifier/非限定动词修饰语

A reduced non-finite verbal modifier is a participial or infinitive form of a verb heading a phrase (which may have some arguments, roughly like a VP). These are used to modify the meaning of an NP or another verb. They are not core arguments of a verb or full finite relative clauses. / 非限定动词修饰语是动词的分词或不定式形式所引导的短语（可能带有若干论元，大体相当于一个 VP），用来修饰名词短语或另一个动词的意义；它既不是动词的核心论元，也不是完整的限定性关系从句。

例：*Points to establish are ...*（有待确立的要点……）——*establish* 是 *points* 的 vmod；*I don't have anything to say to you*（我没有什么要对你说的）——*say* 是 *anything* 的 vmod；*Truffles picked during the spring are tasty*（春天采摘的松露很美味）——*picked* 是 *truffles* 的 vmod；*Bill tried to shoot, demonstrating his incompetence*（比尔试图开枪，显示出他的无能）——*demonstrating* 是 *shoot* 的 vmod。

"Points to establish are . . . "	*vmod*(points, establish)
"I don't have anything to say to you"	*vmod*(anything, say)
"Truffles picked during the spring are tasty"	*vmod*(truffles, picked)
"Bill tried to shoot, demonstrating his incompetence"	*vmod*(shoot, demonstrating)

nn: noun compound modifier/复合名词修饰

A noun compound modifier of an NP is any noun that serves to modify the head noun. / 名词复合修饰语是名词短语中用来修饰中心名词的任何名词。

例：*Oil price futures*（石油价格期货）——*oil* 和 *price* 都是 *futures* 的 nn（复合名词修饰）。

"Oil price futures"	*nn*(futures, oil)
	nn(futures, price)

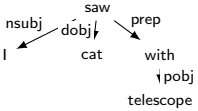
prep: prepositional modifier/介词修饰

A prepositional modifier of a verb, adjective, or noun is any prepositional phrase that serves to modify the meaning of the verb, adjective, noun, or even another preposition. / 介词修饰语是修饰动词、形容词或名词的任何介词短语，用来修饰动词、形容词、名词乃至另一个介词的意义。

例：*I saw a cat in a hat*（我看见一只戴帽子的猫）——*in* 是 *cat* 的 prep；*I saw a cat with a telescope*（我用望远镜看见一只猫）——*with* 是 *saw* 的 prep；*He is responsible for meals*（他负责膳食）——*for* 是 *responsible* 的 prep。

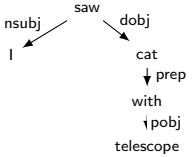
"I saw a cat in a hat"	*prep*(cat, in)
"I saw a cat with a telescope"	*prep*(saw, with)
"He is responsible for meals"	*prep*(responsible, for)

I saw a cat with a telescope.



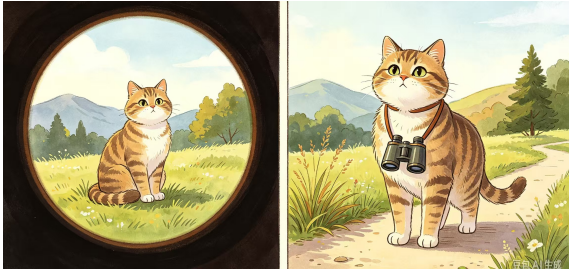
解读 A：*with a telescope* 修饰 **saw**

“我用望远镜看见了那只猫。”



解读 B：*with a telescope* 修饰 **cat**

“我看见了一只带着望远镜的猫。”



root: root

The root grammatical relation points to the root of the sentence. A fake node “ROOT” is used as the governor. The ROOT node is indexed with “0”, since the indexation of real words in the sentence starts at 1. / 根关系指向句子的根节点。系统使用一个虚拟节点“ROOT”作为支配词。ROOT 节点的编号为 0，因为句中真实词的编号从 1 开始。

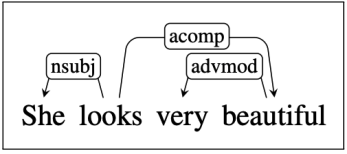
例：*I love French fries*（我爱炸薯条）——root(ROOT, love)；*Bill is an honest man*（比尔是个诚实的人）——root(ROOT, man)（系动词 is 之后，根节点是表语 man）。

“I love French fries.”	*root*(ROOT, love)
“Bill is an honest man”	*root*(ROOT, man)

Stanford Dependency Tree/斯坦福依存树: 更多的依存关系

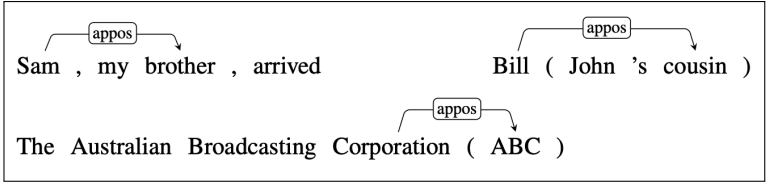
acomp: adjectival complement

An adjectival complement of a verb is an adjectival phrase that functions as the complement (like an object of the verb).



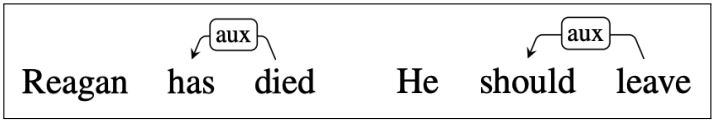
appos: appositional/同位词 modifier

An appositional modifier of an NP is an NP immediately to the right of the first NP that serves to define or modify that NP. It includes parenthesized examples, as well as defining abbreviations in one of these structures.



aux: auxiliary

An auxiliary of a clause is a non-main verb of the clause, e.g., a modal auxiliary, or a form of “be”, “do” or “have” in a periphrastic tense



amod: adjectival modifier

An adjectival modifier of an NP is any adjectival phrase that serves to modify the meaning of the NP.

"Sam eats red meat"	*amod*(meat, red)
"Sam took out a 3 million dollar loan"	*amod*(loan, dollar)
"Sam took out a $ 3 million loan"	*amod*(loan, $)

agent: agent/施事

An agent is the complement of a passive verb which is introduced by the preposition "by" and does the action. This relation only appears in the collapsed dependencies, where it can replace prep by, where appropriate. It does not appear in basic dependencies output.

"The man has been killed by the police"	*agent*(killed, police)
"Effects caused by the protein are important"	*agent*(caused, protein)

auxpass: passive auxiliary

A passive auxiliary of a clause is a non-main verb of the clause which contains the passive information.

"Kennedy has been killed"	*auxpass*(killed, been) *aux*(killed,has)
"Kennedy was/got killed"	*auxpass*(killed, was/got)

dobj: direct object

The direct object of a VP is the noun phrase which is the (accusative) object of the verb.

"She gave me a raise"	*dobj*(gave, raise)
"They win the lottery"	*dobj*(win, lottery)

cop: copula/系动词

A copula is the relation between the complement of a copular verb and the copular verb.

"Bill is big"	*cop*(big, is)
"Bill is an honest man"	*cop*(man, is)

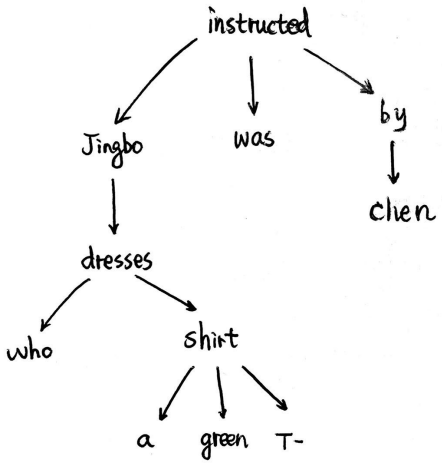
Outline I

1. 1. 实体识别 3

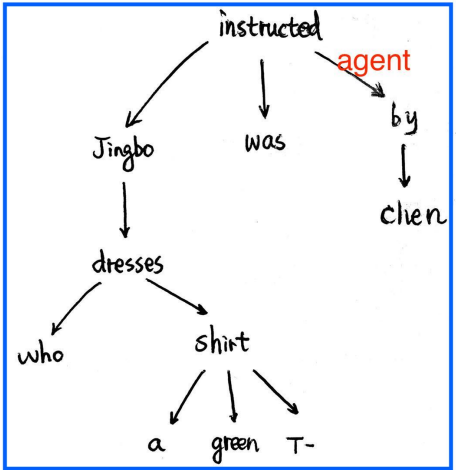
2. 2. 依存关系 Dependency relation 14
   - 2.1 依存关系在语言学标注中的层级位置 15
   - 2.2 斯坦福依存树 / Stanford Dependency Tree 18
   - 2.3 一个例句 37
   - 2.4 进阶1：主谓宾关系抽取（SVO） 52
   - 2.5 进阶2：最短依存路径（SDP, Shortest Dependency Path） 57

3. 3. 语义角色与深层语义 61

4. 4. 代码实现 78



"Jingbo who dresses a green T-shirt was instructed by Chen."

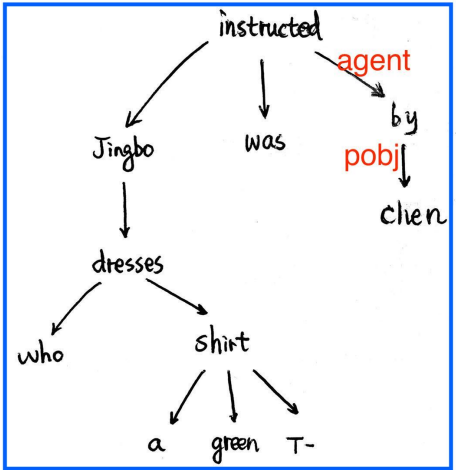


agent: agent

An agent is the complement of a passive verb which is introduced by the preposition “by” and does the action.

“The man has been killed by the police” agent(killed, police)

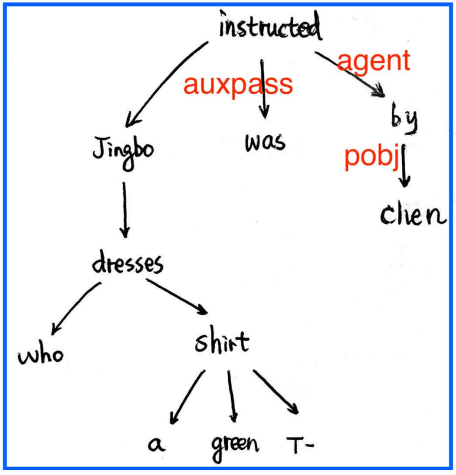
Agent, 施事



pobj: object of a preposition

The object of a preposition is the head of a noun phrase following the preposition, or the adverbs “here” and “there”.

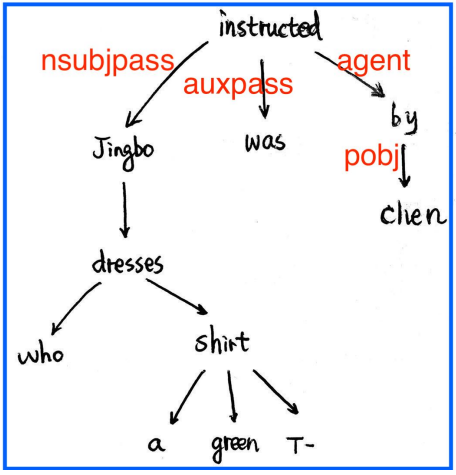
“I sat on the chair” pobj(on, chair)



auxpass: passive auxiliary

A passive auxiliary of a clause is a non-main verb of the clause which contains the passive information.

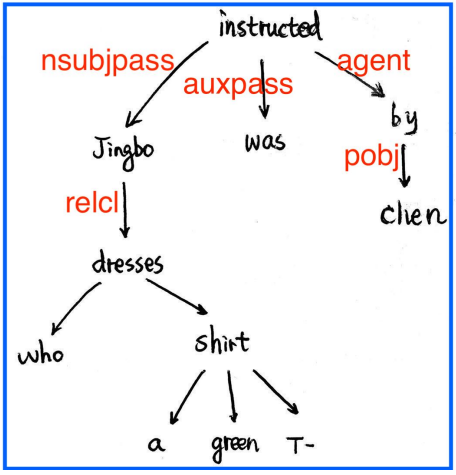
“Kennedy has been killed”
auxpass(killed, been)



nsubjpass: passive nominal subject

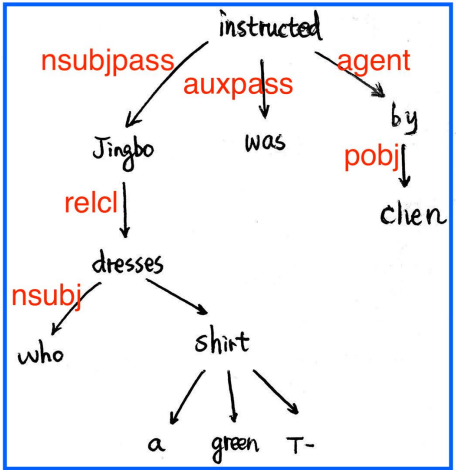
A passive nominal subject is a noun phrase which is the syntactic subject of a passive clause.

“Dole was defeated by Clinton”
nsubjpass(defeated, Dole)



relcl : relative clause

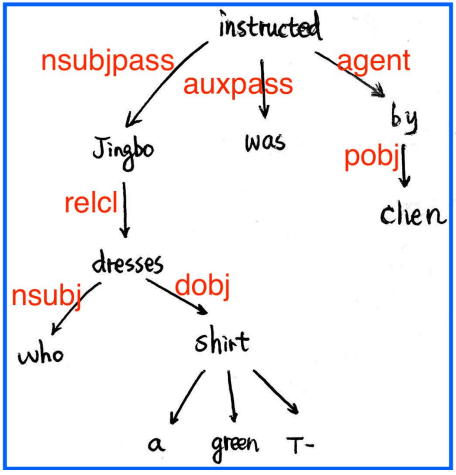
A relative clause (relcl) modifies the head of a noun phrase.



nsubj: nominal subject

A nominal subject is a noun phrase which is the syntactic subject of a clause. The governor of this relation might not always be a verb, which can be an adjective or noun.

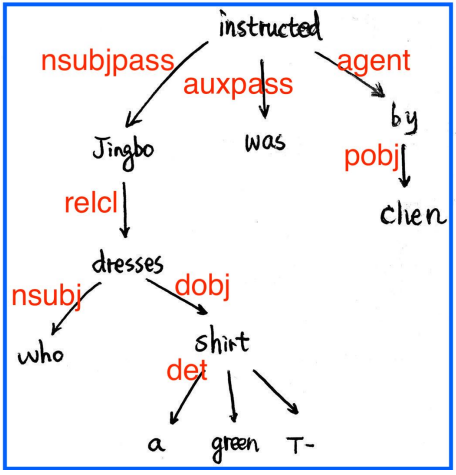
“Clinton defeated Dole”
nsubj(defeated, Clinton)



dobj: direct object

The direct object of a VP is the noun phrase which is the (accusative) object of the verb.

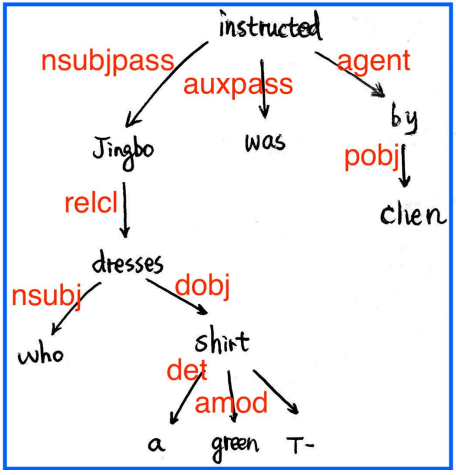
“She gave me a raise” dobj(gave, raise)



det: determiner

A determiner is the relation between the head of an NP and its determiner.

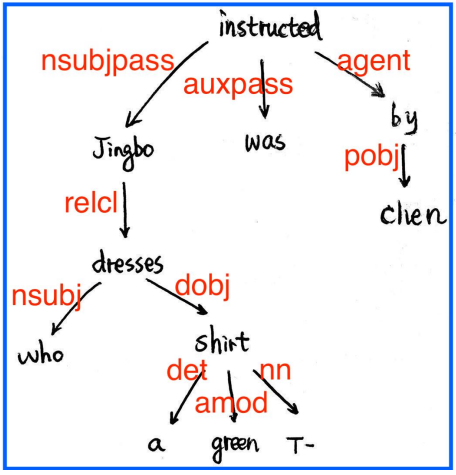
“The man is here” det(man, the)



amod: adjectival modifier

An adjectival modifier of an NP is any adjectival phrase that serves to modify the meaning of the NP.

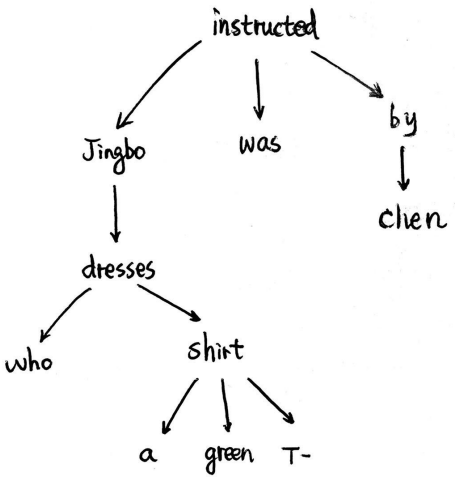
“Sam eats red meat” amod(meat, red)

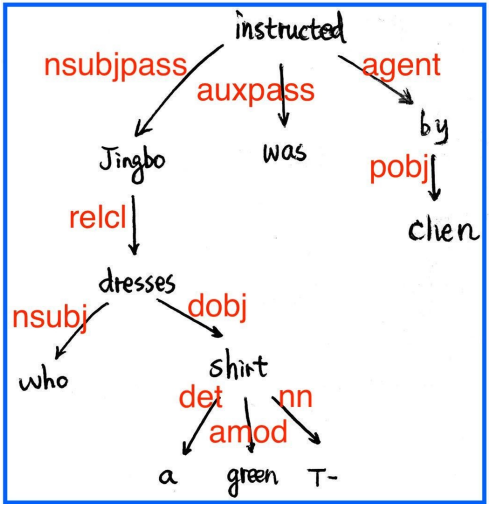


nn: noun compound modifier

A noun compound modifier of an NP is any noun that serves to modify the head noun.

“Oil price futures” nn(futures, oil) nn(futures, price)


instructed
Jingbo
was
by
Chen
dresses
who
shirt
a
green
T-


instructed
nsubjpass
auxpass
agent
Jingbo
was
by
pobj
Chen
relcl
dresses
nsubj
dobj
who
shirt
det
nn
amod
a
green
T-

水稻例句：OsDREB1A improves drought tolerance in rice.

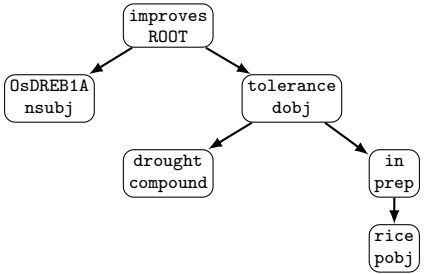
课堂思考

请你仿照前面例句，推导这句话的依存树：

1. 为每个 token 确定它的 `head`（支配词）与 `dep`（依存关系标签）；
2. 画出整棵依存树，标注每条边的关系标签。

提示：参照 Stanford typed dependencies（如 `nsubj`、`dobj`、`compound`、`prep`、`pobj`、`ROOT` 等）。

OsDREB1A improves drought tolerance in rice.



token	head	dep	说明
OsDREB1A	improves	nsubj	主语（基因）
improves	improves	ROOT	谓语，整句核心
drought	tolerance	compound	复合修饰语
tolerance	improves	dobj	宾语（性状）
in	tolerance	prep	介词修饰
rice	in	pobj	介词宾语

Outline I

1. 1. 实体识别 3

2. 2. 依存关系 Dependency relation 14
   - 2.1 依存关系在语言学标注中的层级位置 15
   - 2.2 斯坦福依存树 / Stanford Dependency Tree 18
   - 2.3 一个例句 37
   - 2.4 进阶1：主谓宾关系抽取（SVO） 52
   - 2.5 进阶2：最短依存路径（SDP, Shortest Dependency Path） 57

3. 3. 语义角色与深层语义 61

4. 4. 代码实现 78

请思考：如果在包含两个实体的一个句子中，两个实体恰好位于主语和宾语，该主谓宾结构能否提供
(实体-1，“影响”,实体-2)
这样的三元组（Triple）信息？

OsDREB1A improves drought tolerance in rice.

实体：OsDREB1A（基因）、drought tolerance（性状）

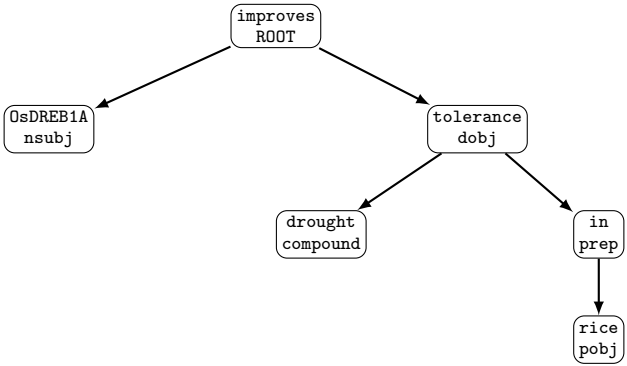
譬如：从上述句子可抽取

(OsDREB1A, improves, drought tolerance).

请思考：如何从依存关系中抽取SVO结构？

关键：找到 nsubj（主语）与 dobj（宾语）的 head 是否为同一个动词（谓语）。

依存树展示 SVO 结构：


improves
ROOT
OsDREB1A
nsubj
tolerance
dobj
drought
compound
in
prep
rice
pobj

关键语法关系说明

1. **主谓关系 (`nsubj`)：`OsDREB1A` 的 head 是 `improves`，表示基因是动作发出者。**
2. **动宾关系 (`dobj`)：`tolerance` 的 head 是 `improves`，表示性状是动作承受者。**
3. **复合词关系 (`compound`)：`drought` 的 head 是 `tolerance`，构成完整性状 `drought tolerance`。**
4. **介词短语 (`prep + pobj`)：`in rice` 修饰 `tolerance`，表示地点范围。**

生物医学语义分析
从依存解析结果可提取：

- **主体 (Agent)：OsDREB1A**
- **作用 (Action)：improves**
- **客体 (Target)：drought tolerance**

这种主—谓—宾结构准确表达了“OsDREB1A 基因改善水稻耐旱性”的核心科学论断。

Outline I

1. 1. 实体识别 3

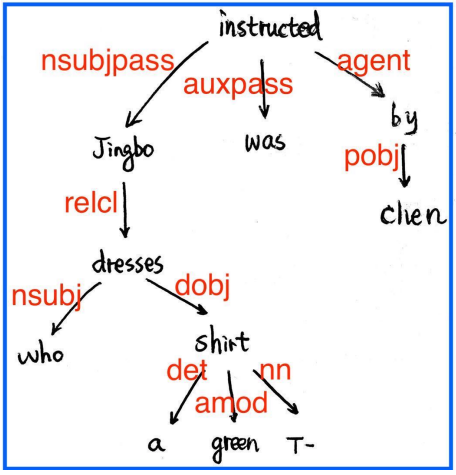
2. 2. 依存关系 Dependency relation 14
   - 2.1 依存关系在语言学标注中的层级位置 15
   - 2.2 斯坦福依存树 / Stanford Dependency Tree 18
   - 2.3 一个例句 37
   - 2.4 进阶1：主谓宾关系抽取（SVO） 52
   - 2.5 进阶2：最短依存路径（SDP, Shortest Dependency Path） 57

3. 3. 语义角色与深层语义 61

4. 4. 代码实现 78

从依存树到最短依存路径：

1. 依存树（dependency tree）：
   语义依存解析把句子拆解为树形结构，关注的是在句中位置较远但在语法上紧密相连的词语。
2. 最短依存路径（SDP）：
   在无向依存图上找到连接两个目标实体的最短语法路径。



When focusing on head entity (‘Jingbo’) and tail entity (‘Chen’).

Short Dependency Path (SDP): ‘Jingbo-instructed-by-Chen’

Length:
3

Triple in the manner of KG:
(h,r,t)=(Jingbo,instructed,Chen)

例句：”Jingbo who dresses a green T-shirt was instructed by Chen.”

最短依存路径的应用场景：

1. 关系抽取中的特征工程（预深度学习时代的主流做法）；
2. 知识图谱中的三元组抽取；
3. 作为神经网络的输入，利用SDP所携带的语法语义信息。

Outline I

1. 1. 实体识别 3
   - 1.1 实体识别性能的评估 4
   - 1.2 提升实体识别与信息抽取的方向 12
2. 2. 依存关系 Dependency relation 14
   - 2.1 依存关系在语言学标注中的层级位置 15
   - 2.2 斯坦福依存树 / Stanford Dependency Tree 18
   - 2.3 一个例句 37
   - 2.4 进阶1：主谓宾关系抽取（SVO） 52
   - 2.5 进阶2：最短依存路径（SDP, Shortest Dependency Path） 57
3. 3. 语义角色与深层语义 61
   - 3.1 语义角色的定义 62
   - 3.2 医药文献知识挖掘：语义角色对深层语义的建模 67
   - 3.3 语料库和评测任务 73
4. 4. 代码实现 78
   - 4.1 最短依存路径——SpaCy for Dependency Tree & Networkx 83
   - 4.2 主谓宾关系抽取（SVO）——对应 2.4 进阶1 91
   - 4.3 从主谓宾出发：知识条目的标准化处理 104

Outline I

1. 1. 实体识别 3

2. 2. 依存关系 Dependency relation 14

3. 3. 语义角色与深层语义 61
   - 3.1 语义角色的定义 62
   - 3.2 医药文献知识挖掘：语义角色对深层语义的建模 67
   - 3.3 语料库和评测任务 73

4. 4. 代码实现 78

语义角色(Semantic Roles)指有关语言成分的所指在语句所表达的事件中所扮演的参与者角色（Participant Roles）。

从某种意义上说，语义角色是语言学家对句子中有关结构成分之间的意义关系的一种分类。这种分类的粗细程度，可以因语言学家的认识或具体的应用目标的不同而不同：从非常专门的到非常一般的，从而形成了不同等级的语义角色集合。

语义角色标注（semantic role labeling）也叫做语义分析（semantic parsing），是用来识别句子中谓语的 predicate-argument 结构，比如 “Who did what to whom, where and when?”，在给定谓语的情况下标记语义角色[1]。

语义角色标注（semantic role labeling）也叫做语义分析（semantic parsing），是用来识别句子中谓语的 predicate-argument 结构，比如 “Who did what to whom, where and when?”，在给定谓语的情况下标记语义角色。

- 致事（Cause）：动作发生的客观原因。
  - 某种致使性事件的引起者（北京大学）
  - ...what caused the action to occur in the first place; not for what, rather because of what (e.g., Because Clyde was hungry, he ate the cake.). （宾州大学）

语义角色标注（semantic role labeling）也叫做语义分析（semantic parsing），是用来识别句子中谓语的 predicate-argument 结构，比如 “Who did what to whom, where and when?”，在给定谓语的情况下标记语义角色。

- 致事（Cause）：动作发生的客观原因。
  - 某种致使性事件的引起者（北京大学）
  - ...what caused the action to occur in the first place; not for what, rather because of what (e.g., Because Clyde was hungry, he ate the cake.). （宾州大学）
- 主事（Theme）：性质、状态或变化性事件的主体。
  - 性质、状态等无施动、感知性的主体（北京大学）
  - ... undergoes the action but does not change its state (e.g., We believe in one God. I have two children. I put the book on the table. He gave the gun to the police officer.) (Sometimes used interchangeably with patient/施事.) （宾州大学）

[1] https://zh.wikipedia.org/wiki/\CTEX@char@nnn{232}{175}{173}\CTEX@char@nnn{228}{185}{137}\CTEX@char@nnn{232}{167}{146}\CTEX@char@nnn{232}{137}{178}

Outline I

1. 1. 实体识别 3

2. 2. 依存关系 Dependency relation 14

3. 3. 语义角色与深层语义 61
   - 3.1 语义角色的定义 62
   - 3.2 医药文献知识挖掘：语义角色对深层语义的建模 67
   - 3.3 语料库和评测任务 73

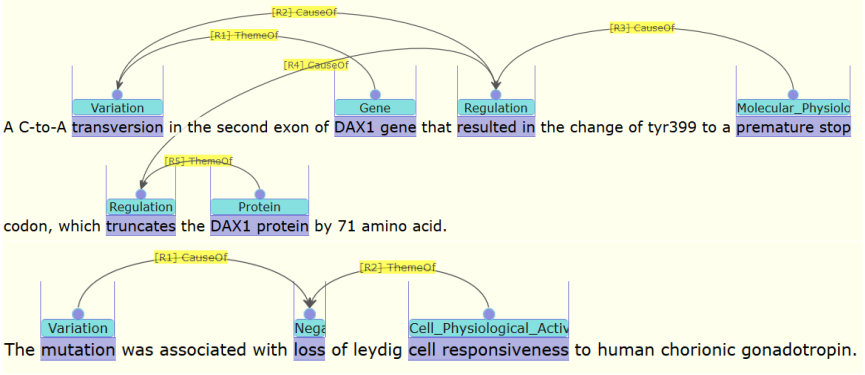
4. 4. 代码实现 78

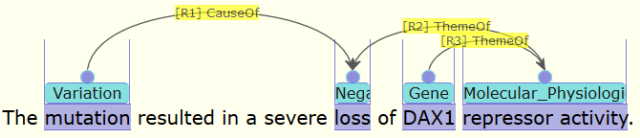
语义角色所揭示的深层语义：

在某种外界条件影响下，基因XXX发生了某种突变，导致失去或者获得功能，这与某种疾病的发病原理有关。

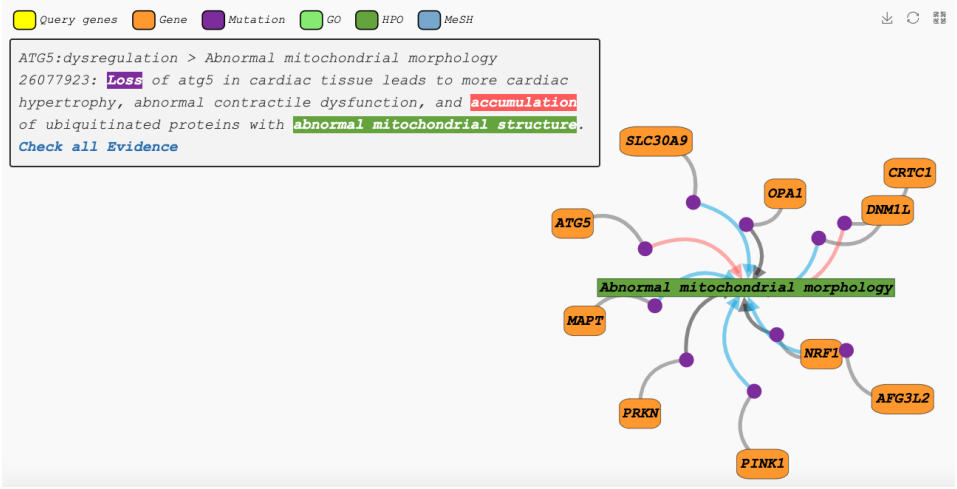
vs.

"Who did what to whom, where and when?"


[R2] CauseOf
[R1] ThemeOf
[R3] CauseOf
[R4] CauseOf
Variation
Gene
Regulation
Molecular_Physiolo
A C-to-A transversion in the second exon of DAX1 gene that resulted in the change of tyr399 to a premature stop
[R5] ThemeOf
Regulation
Protein
codon, which truncates the DAX1 protein by 71 amino acid.
[R1] CauseOf
[R2] ThemeOf
Variation
Nega
Cell_Physiological_Activ
The mutation was associated with loss of leydig cell responsiveness to human chorionic gonadotropin.



人工标注的语料库-可视化界面


Query genes
Gene
Mutation
GO
HPO
MeSH
ATG5:dysregulation > Abnormal mitochondrial morphology
26077923: Loss of atg5 in cardiac tissue leads to more cardiac hypertrophy, abnormal contractile dysfunction, and accumulation of ubiquitinated proteins with abnormal mitochondrial structure.
Check all Evidence
SLC30A9
OPA1
CRTC1
DNM1L
ATG5
Abnormal mitochondrial morphology
MAPT
NRF1
AFG3L2
PRKN
PINK1

自动化挖掘结果——可视化界面。点击链接可查看更多： http://lit-evi.hzau.edu.cn/AD-LitPathoNet/visualization?QueryPhenotype=Abnormal%20mitochondrial%20morphology

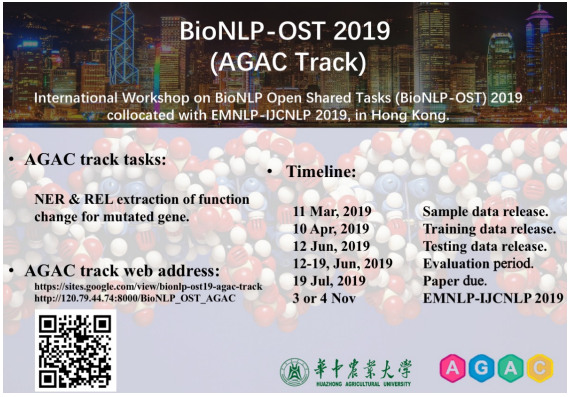
Outline I

1. 1. 实体识别 3

2. 2. 依存关系 Dependency relation 14

3. 3. 语义角色与深层语义 61
   - 3.1 语义角色的定义 62
   - 3.2 医药文献知识挖掘：语义角色对深层语义的建模 67
   - 3.3 语料库和评测任务 73

4. 4. 代码实现 78


BioNLP-OST 2019
(AGAC Track)
International Workshop on BioNLP Open Shared Tasks (BioNLP-OST) 2019
collocated with EMNLP-IJCNLP 2019, in Hong Kong.
• AGAC track tasks:
NER & REL extraction of function change for mutated gene.
• AGAC track web address:
https://sites.google.com/view/bionlp-ost19-agac-track
http://120.79.44.74:8000/BioNLP_OST_AGAC
• Timeline:
11 Mar. 2019 Sample data release.
10 Apr, 2019 Training data release.
12 Jun, 2019 Testing data release.
12-19, Jun, 2019 Evaluation period.
19 Jul, 2019 Paper due.
3 or 4 Nov EMNLP-IJCNLP 2019
華中農業大學
HUAZHONG AGRICULTURAL UNIVERSITY
AGAC



The International Workshop on BioNLP Open Shared Tasks (BioNLP-OST) 2019 will be collocated with **EMNLP-IJCNLP 2019** either on 3rd or 4th of November, in Hong Kong.

https://2019.bionlp-ost.org/home



http://lit-evi.hzau.edu.cn/AGAC-CHIP2022/



面向“基因–疾病”关联机理的科学文献挖掘任务（AGAC）

任务描述

在海量科学文献中，“基因–疾病”的关联机理通过突变、基因等系列分子对象及其触发词联得描述，自然语言处理为自动挖掘这一隐性知识条目提供了可能，亦为健康医学信息的自动化处理提供解决方案。

本任务来自CHIP2022学术评测任务一，研究人员可以访问官网申请数据：http://lit-evi.hzau.edu.cn/AGAC-CHIP2022/

任务说明

本任务包括三个子任务：1、触发词实体识别；2、语义角色识别，3、“基因，功能变化，疾病”知识发现：

- 任务1是传统意义下的命名实体识别，用以识别十二类与“基因–疾病”有关的分子对象及其触发词实体，包括Var, MPA, Interaction, Pathway, CPA, Reg, PosReg, NegReg, Disease, Gene, Protein, 和Enzyme。
- 任务2是一个语义角色识别任务，语义角色包括ThemeOf,和CauseOf。该任务捕捉实体之间的语义依赖关系，用以构建“基因–疾病”关联。
- 任务3是针对“基因–疾病”的关联机理的预测任务，可依赖任务1和任务2的关联关系，预测“基因–功能改变–疾病”的关联机理。此处，功能改变包含4种对突变基因的语义描述：功能丧失(LOF)、功能获得(GOF)、功能调节(REG)、和功能的复合变化(COM)。
  任务对象定义来源于AGAC（Active Gene Annotation Corpus）活跃基因注释语料库，该语料库主要用以挖掘突变引起的“基因–疾病”关联机理。AGAC语料库包括四类分子对象、八类触发词实体，两个语义角色，以及四种用以描述“基因–疾病”关联机理的功能变化。

说明：AGAC中只有同时涉及特定突变和生物学功能或疾病的句子才会被注释。

任务一识别对象

分子对象：

- Disease(疾病)
- Gene(基因)

面向“基因–疾病”关联机理的...
任务描述
任务说明
数据格式
评测指标
评测数据
数据集信息
Github
数据集引用文章
系统排名
如何下载
如何提交
更新日志
Github
引用
参考文献
社区贡献
教学实践
致谢
FAQ

阿里云 | TIANCHI 天池

https://tianchi.aliyun.com/dataset/95414

Outline I

1. 1. 实体识别 3
   - 1.1 实体识别性能的评估 4
   - 1.2 提升实体识别与信息抽取的方向 12
2. 2. 依存关系 Dependency relation 14
   - 2.1 依存关系在语言学标注中的层级位置 15
   - 2.2 斯坦福依存树 / Stanford Dependency Tree 18
   - 2.3 一个例句 37
   - 2.4 进阶1：主谓宾关系抽取（SVO） 52
   - 2.5 进阶2：最短依存路径（SDP, Shortest Dependency Path） 57
3. 3. 语义角色与深层语义 61
   - 3.1 语义角色的定义 62
   - 3.2 医药文献知识挖掘：语义角色对深层语义的建模 67
   - 3.3 语料库和评测任务 73
4. 4. 代码实现 78
   - 4.1 最短依存路径——SpaCy for Dependency Tree & Networkx 83
   - 4.2 主谓宾关系抽取（SVO）——对应 2.4 进阶1 91
   - 4.3 从主谓宾出发：知识条目的标准化处理 104

例句与问题暴露

例句： We found that OsDREB1A improves drought tolerance.

问题： 用第二版 `extract_svo` 处理该句，输出 `None`——找不到 SVO。

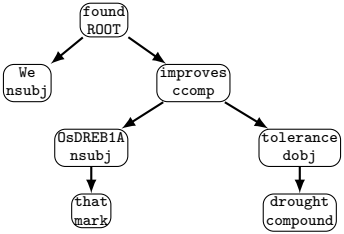
原因分析

```
doc = nlp("We found that OsDREB1A improves drought tolerance.")

for t in doc:
    print(f"{t.dep_:12} {t.head.text:12} {t.text}")
# nsubj        found        We
# ROOT         found        found
# mark         improves     that
# nsubj        improves     OsDREB1A
# ccomp        found        improves
# compound     tolerance    drought
# dobj         improves     tolerance
# punct        found        .
```

found 是 ROOT，但真正的 SVO 在 that 从句中：improves 的 dep_ 是 ccomp（从句补语），而非 ROOT。第二版代码只检查 t.dep_ == "ROOT"，因此遗漏。

依存树可视化



修正逻辑：扩大动词搜索范围

```python
def extract_svo_v3(doc):
    for t in doc:
        # 不仅检查，还检查从句动词 ROOT
        if t.dep_ not in ("ROOT", "ccomp", "relcl", "advcl"):
            continue
        subj = [c for c in t.children if c.dep_ == "nsubj"]
        obj  = [c for c in t.children if c.dep_ in ("dobj", "obj")]
        if subj and obj:
            def span(x):
                mods = [c.text for c in x.lefts
                        if c.dep_ in ("compound", "amod")]
                return "-".join(mods + [x.text])
            return (span(subj[0]), t.text, span(obj[0]))
    return None

print(extract_svo_v3(doc))
# ('OsDREB1A', 'improves', 'drought tolerance')
```

第三版与第二版的唯一差异：第 3 行 `if t.dep_ == "ROOT"` 改为 `if t.dep_ not in ("ROOT", "ccomp", "relcl", "advcl")`。

代码演化总结

- **第一版**：ROOT 下找 nsubj + dobj，宾语带 compound 左修饰

- **第二版**：span() 对主语、宾语对称处理，支持 compound / amod
- **第三版**：动词范围从 ROOT 扩展到 ccomp / relcl / advcl，覆盖从句

Outline I

1. 1. 实体识别 3
2. 2. 依存关系 Dependency relation 14
3. 3. 语义角色与深层语义 61
4. 4. 代码实现 78
   - 4.1 最短依存路径——SpaCy for Dependency Tree & Networkx 83
   - 4.2 主谓宾关系抽取（SVO）——对应 2.4 进阶1 91
   - 4.3 从主谓宾出发：知识条目的标准化处理 104

Installation of Spacy and Networkx

```
> pip install spacy networkx
> python -m spacy download en_core_web_sm
> python -c "import spacy; print(spacy.__version__)"
# 3.8.18
```

说明：`spacy download` 会自动匹配已安装 spaCy 的模型版本，无需手工指定版本号；离线环境可下载模型后执行 `pip install en_core_web_sm-3.8.0-py3-none-any.whl`。

获取依存树（Obtain dependency tree）

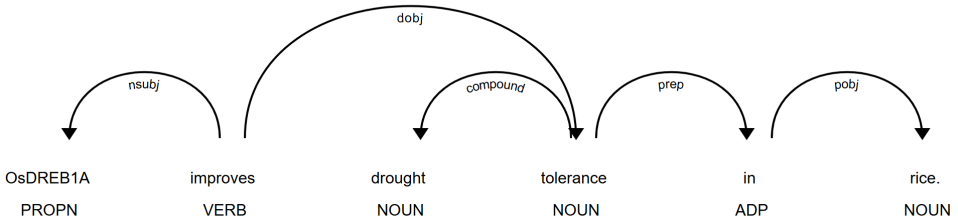
```
import spacy
nlp = spacy.load("en_core_web_sm")

doc = nlp("OsDREB1A improves drought tolerance in rice.")
for token in doc:
    print(f"{token.dep_}({token.head.text},{token.text})")
#nsubj(improves, OsDREB1A)
#ROOT(improves, improves)
#compound(tolerance, drought)
#dobj(improves, tolerance)
#prep(tolerance, in)
#pobj(in, rice)
#punct(improves, .)
```

输出沿用第 2 讲的 dep(head, token) 记号：每行依次给出关系标签、支配词与从属词。

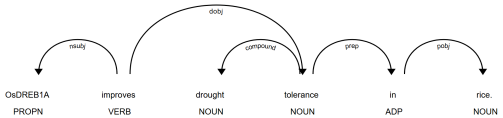
可视化依存树（Visualize dependency tree）

```
1 from spacy import displacy
2 displacy.render(doc, style="dep", jupyter=True)   # 在 Jupyter 中内嵌显示
3 # 也可在浏览器中打开：displacy.serve(doc, style="dep")
```

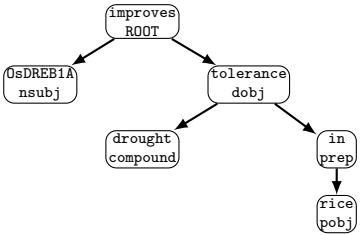


两种画法对照：spaCy 输出 vs. Stanford风格依存树

spaCy displacy 输出



Stanford风格依存树（箭头从上往下）



对照要点：两图节点相同、依存关系一致；左图箭头从 head 指向 dep（弧形标注），右图箭头同样从 head 指向 dep（直线从上往下）。

水稻例句：把依存树存入图

```
import networkx as nx

doc = nlp("OsDREB1A-improves-drought-tolerance-in-rice.")

G = nx.Graph()
for token in doc:
    G.add_node(token.i, text=token.text)
for token in doc:
    for child in token.children:
        if token.dep_ == "punct" or child.dep_ == "punct":
            continue
        G.add_edge(child.i, token.i, dep=child.dep_)

# 节点用 token.，避免同名同形词被合并成一个节点i
print(list(G.edges(data=True)))
#[(0,1,{'dep':'nsubj'}), (1,3,{'dep':'dobj'}), (2,3,{'dep':'compound'}),
# (3,4,{'dep':'prep'}), (4,5,{'dep':'pobj'})]
```

代码中的 Tokens 与 punct 说明

Tokens 列表

索引	文本
0	OsDREB1A
1	improves
2	drought
3	tolerance
4	in
5	rice
6	.

句号也是 spaCy 解析出的一个 token（索引 6），关系标签为 punct。

完整依存输出（含标点）

```
for token in doc:
    print(f"{token.dep_}({token.head.text},-{token.text})")
#nsubj(improves, OsDREB1A)
#ROOT(improves, improves)
#compound(tolerance, drought)
#dobj(improves, tolerance)
#prep(tolerance, in)
#pobj(in, rice)
#punct(improves, .)
```

第 2 讲手绘 TikZ 图省略了标点；代码中 `if token.dep_ == "punct"` 将句号过滤，使 NetworkX 图与理论图保持一致。

水稻例句：最短路径

```
gene, trait = doc[0], doc[3]          # OsDREB1A, tolerance
path = nx.shortest_path(G, source=gene.i, target=trait.i)
print([doc[i].text for i in path])
#['OsDREB1A', 'improves', 'tolerance']
print([G[a][b]['dep'] for a, b in zip(path, path[1:])])
#['nsubj', 'dobj']
```

Outline I

1. 1. 实体识别 3
2. 2. 依存关系 Dependency relation 14
3. 3. 语义角色与深层语义 61
4. 4. 代码实现 78
   - 4.1 最短依存路径——SpaCy for Dependency Tree & Networkx 83
   - 4.2 主谓宾关系抽取（SVO）——对应 2.4 进阶1 91
   - 4.3 从主谓宾出发：知识条目的标准化处理 104

从依存树抽取三元组

```python
def extract_svo(doc):
    for t in doc:
        if t.dep_ != "ROOT":
            continue
        subj = [c for c in t.children if c.dep_ == "nsubj"]
        obj  = [c for c in t.children if c.dep_ in ("dobj", "obj")]
        if subj and obj:
            def span(x):
                mods = [c.text for c in x.lefts
                        if c.dep_ in ("compound", "amod")]
                return "-".join(mods + [x.text])
            return (span(subj[0]), t.text, span(obj[0]))

print(extract_svo(doc))
#('OsDREB1A', 'improves', 'drought tolerance')
```

对应 2.4 理论：找 nsubj 与 dobj 的 head 是否为同一动词；局限：被动句返回 None，仅覆盖主动宾结构。

例句与适用条件

例句： OsDREB1A improves drought tolerance in rice.

适用条件：

1. 句子为**主动宾结构**：主语（nsubj）与宾语（dobj/obj）的 head 是同一动词（ROOT）。
2. 主语或宾语含**左修饰**（compound/amod），如 drought tolerance。

逐步模拟输出

```
doc = nlp("OsDREB1A-improves-drought-tolerance-in-rice.")

# 1. 遍历，找到 token ROOT
for t in doc:
    if t.dep_ == "ROOT":
        print(t.text)
# improves

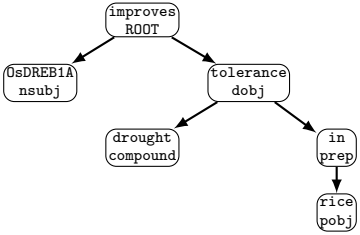
# 2. 找 ROOT 的子节点中 nsubj 与 dobj
subj = [c for c in t.children if c.dep_ == "nsubj"]   # [OsDREB1A]
obj  = [c for c in t.children if c.dep_ in ("dobj", "obj")]
                                                      # [tolerance]

# 3. 对宾语收集左修饰 compound/amod
mods = [c.text for c in obj[0].lefts
        if c.dep_ in ("compound", "amod")]
print(mods)
# ['drought']

# 4. 拼接并返回三元组
print((subj[0].text, t.text, "-".join(mods + [obj[0].text])))
# ('OsDREB1A', 'improves', 'drought tolerance')
```

若条件不满足（如被动句 nsubjpass 结构），subj 或 obj 为空，函数返回 None。

依存树可视化



例句与适用条件

例句： OsDREB1A activates downstream genes.

适用条件：

1. 句子为**主动宾结构**：nsubj 与 dobj 的 head 是同一动词（ROOT）。
2. 宾语含**左修饰**（amod），如 downstream genes。

逐步模拟输出

```
doc = nlp("OsDREB1A-activates-downstream-genes.")

# 1. 遍历，找到 token ROOT
for t in doc:
    if t.dep_ == "ROOT":
        print(t.text)
# activates

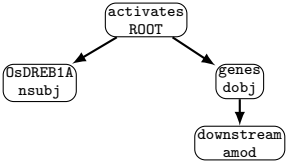
# 2. 找 ROOT 的子节点中 nsubj 与 dobj
subj = [c for c in t.children if c.dep_ == "nsubj"]   # [OsDREB1A]
obj  = [c for c in t.children if c.dep_ in ("dobj", "obj")]
                                                      # [genes]

# 3. 对宾语收集左修饰 compound/amod
mods = [c.text for c in obj[0].lefts
        if c.dep_ in ("compound", "amod")]
print(mods)
# ['downstream']

# 4. 拼接并返回三元组
print((subj[0].text, t.text, "-".join(mods + [obj[0].text])))
# ('OsDREB1A', 'activates', 'downstream-genes')
```

与水稻句对比：drought tolerance 走 compound 分支，downstream genes 走 amod 分支；span() 函数同时覆盖两者。

依存树可视化



例句与适用条件

例句： OsDREB1A overexpression enhances drought tolerance.

适用条件：

1. 句子为**主动宾结构**：`nsubj` 与 `dobj` 的 head 是同一动词（`ROOT`）。
2. **主语含左修饰**（`compound`），如 `OsDREB1A overexpression`。

逐步模拟输出

```
doc = nlp("OsDREB1A-overexpression-enhances-drought-tolerance.")

# 1. 遍历，找到 token ROOT
for t in doc:
    if t.dep_ == "ROOT":
        print(t.text)
# enhances

# 2. 找 ROOT 的子节点中 nsubj 与 dobj
subj = [c for c in t.children if c.dep_ == "nsubj"]   # [overexpression]
obj  = [c for c in t.children if c.dep_ in ("dobj", "obj")]
                                                      # [tolerance]

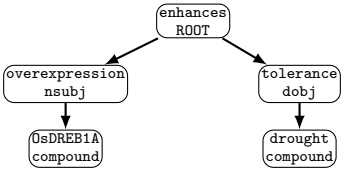
# 3. 对主语和宾语分别收集左修饰 compound/amod
def span(x):
    mods = [c.text for c in x.lefts
            if c.dep_ in ("compound", "amod")]
    return "-".join(mods + [x.text])

print(span(subj[0]))   # OsDREB1A overexpression
print(span(obj[0]))    # drought tolerance

# 4. 拼接并返回三元组
print((span(subj[0]), t.text, span(obj[0])))
# ('OsDREB1A overexpression', 'enhances', 'drought tolerance')
```

与之前例句对比：主语和宾语都可能带左修饰，span() 对两者对称处理。

依存树可视化



依存关系标准文档

- **Stanford Typed Dependencies Manual**
  de Marneffe, M.-C., & Manning, C. D. (2008/2015). Stanford University.
  https://ufal.mff.cuni.cz/~hladka/2015/docs/Stanford-dependencies-manual.pdf
  定义 `nsubj`、`dobj`、`ccomp`、`relcl` 等 50+ 依存关系，spaCy 输出直接对应。
- **Universal Dependencies v2**
  Nivre, J., et al. (2020). https://universaldependencies.org/
  跨语言依存标注标准，spaCy 3.x 默认采用 UD 标签体系。

语言学教材

- **Understanding Syntax** (5th ed.)
  Tallerman, M. (2020). Routledge. DOI: 10.4324/9780429243592
  Chapter 3 "Grammatical relations" 讲解主语、宾语的跨语言定义与识别方法。
- **Speech and Language Processing** (3rd ed. draft, Chapter 19)
  Jurafsky, D., & Martin, J. H. (2024). Stanford University.
  https://web.stanford.edu/~jurafsky/slp3/old_aug24/19.pdf
  依存解析章，含 `nsubj`/`dobj`/`ccomp` 的 NLP 视角定义。

实用分析手册

- **SPOCA Checksheet**
  Lancaster University, Linguistics Department.
  https://www.lancaster.ac.uk/fass/projects/stylistics/topic4a/6spoca.htm
  Subject-Predicate-Object-Complement-Adverbial 五步分析法，适合课堂演示。

Outline I

1. 1. 实体识别 3
2. 2. 依存关系 Dependency relation 14
3. 3. 语义角色与深层语义 61
4. 4. 代码实现 78
   - 4.1 最短依存路径——SpaCy for Dependency Tree & Networkx 83
   - 4.2 主谓宾关系抽取（SVO）——对应 2.4 进阶1 91
   - 4.3 从主谓宾出发：知识条目的标准化处理 104

问题提出

当前 `extract_svo` 返回的是**表面形式**（surface form）：

- `('OsDREB1A', 'improves', 'drought tolerance')`
- `('OsDREB1A', 'activates', 'downstream genes')`
- `('OsDREB1A overexpression', 'enhances', 'drought tolerance')`

核心问题：`drought tolerance` 与 `drought resistance` 是同一性状吗？`tolerance`、`tolerant`、`tolerate` 如何归并？

三步走策略概览

1. **词形还原（Lemmatization）**：统一词形变化，解决"拼写不同但词根相同"
2. **术语映射（Term Mapping）**：对接领域术语表，解决"说法不同但概念相同"
3. **实体链接（Entity Linking）**：对接外部知识库，解决"同名异义、异名同义"的消歧

三步由浅入深，从纯语言学操作逐步过渡到领域知识融合。

为什么需要词形还原？

观察以下生物学句子，它们描述同一事实，但动词形式不同：

- OsDREB1A improves drought tolerance.（一般现在时，陈述普遍事实）
- OsDREB1A improved drought tolerance.（过去时，描述实验结果）
- OsDREB1A improve drought tolerance.（复数主语或虚拟语气）
- OsDREB1A is improving drought tolerance.（现在进行时，强调过程）

若直接以表面形式存储三元组，会得到四个不同的谓语：improves、improved、improve、improving。但实际上它们指向同一动作，知识库中应只保留一个条目。

生物学中的动词屈折实例

生物学动词	屈折形式	例句
express	expresses, expressed, expressing	OsDREB1A expresses in roots.
regulate	regulates, regulated, regulating	OsDREB1A regulates downstream genes.
bind	binds, bound, binding	OsDREB1A binds to DRE element.
enhance	enhances, enhanced, enhancing	OsDREB1A enhances drought tolerance.
inhibit	inhibits, inhibited, inhibiting	OsDREB1A inhibits leaf senescence.

原理：屈折变化与词元

屈折变化（Inflection）

同一词汇为表达语法功能（时态、数、人称）而产生的变形，**不改变词汇的核心语义和词性**。英语属于弱屈折语言，变化主要体现在动词和名词上。

- **动词屈折**：improve（原形）→ improves（第三人称单数现在时）→ improved（过去式/过去分词）→ improving（现在分词）。四个形式共享同一语义"改善"。
- **名词屈折**：gene（单数）→ genes（复数）。两个形式共享同一语义"基因"。
- **形容词/副词**：tolerant（形容词）→ tolerantly（副词），但 tolerance（名词）是派生而非屈折。

词元（Lemma）与词典

词元的定义

词元是词典中收录的**标准引用形式**（citation form），代表一组屈折变化的共同词根。英语词典通常以动词原形、名词单数、形容词原级作为词元。

表面形式	词元	变化类型
improves	improve	第三人称单数
improved	improve	过去式
genes	gene	复数
tolerance	tolerance	无变化（已是词元）
expressed	express	过去式/过去分词
regulates	regulate	第三人称单数
bound	bind	过去式（不规则）
better	good	不规则比较级

spaCy 实现原理

```
# spaCy 的 lemmatizer 组件工作流程：
# 1. 查异常表（lookup）：exceptionswent -> go, better -> good
# 2. 应用规则表（rule）：based-s, -ed, -ing 后缀剥离
# 3. 词性约束（POS）：根据词性选择还原规则constraint

doc = nlp("OsDREB1A-improves-drought-tolerance.")
print([(t.text, t.lemma_, t.pos_) for t in doc])
# [('OsDREB1A', 'OsDREB1A', 'PROPN'),
#  ('improves', 'improve', 'VERB'),
#  ('drought', 'drought', 'NOUN'),
#  ('tolerance', 'tolerance', 'NOUN'),
#  ('.', '.', 'PUNCT')]
```

- improves → improve：动词，规则剥离 -s
- genes → gene：名词，规则剥离 -s
- tolerance → tolerance：名词单数，无需变化
- expressed → express：动词过去分词，规则剥离 -ed
- bound → bind：不规则动词，查异常表还原

词形还原 vs 词干提取

	词形还原（Lemmatization）	词干提取（Stemming）
方法	词典查找 + 规则	启发式截断
结果	合法词汇	可能不是合法词汇
例子	`improved` → `improve`	`improved` → `improv`
例子	`went` → `go`	`went` → `went`（无法处理）

局限：词形相同，语义不同

词形还原只解决**词形层面**的归并。以下短语词形不同、词根也不同，但语义相近：

- `drought tolerance`（耐旱性）—— 植物在干旱条件下维持生长的能力
- `drought resistance`（抗旱性）—— 植物抵抗干旱胁迫的能力
- `drought escape`（避旱性）—— 植物通过提前成熟避开干旱的策略

在植物生物学中，`tolerance` 和 `resistance` 常互换使用，指向同一性状；但 `escape` 是不同机制，不应归并。词形还原无法识别这种语义差异，需要第二步术语映射。

为什么需要术语映射？

以下生物学句子描述同一性状，但使用了不同的术语变体：

- OsDREB1A improves drought tolerance.
- OsDREB1A improves drought resistance.
- OsDREB1A improves drought stress tolerance.
- OsDREB1A enhances salt tolerance.
- OsDREB1A enhances salinity tolerance.
- OsDREB1A enhances salt stress tolerance.

词形还原后分别为 drought tolerance、drought resistance、drought stress tolerance、salt tolerance、salinity tolerance、salt stress tolerance。植物生物学家知道前三者指向同一概念，后三者也指向同一概念。

生物学中的术语变体实例

性状类别	术语变体
耐旱性	drought tolerance, drought resistance, drought stress tolerance, water deficit tolerance
耐盐性	salt tolerance, salinity tolerance, salt stress tolerance, NaCl tolerance
耐寒性	cold tolerance, chilling tolerance, freezing tolerance, low temperature tolerance
耐热性	heat tolerance, heat stress tolerance, high temperature tolerance, thermotolerance
耐病性	disease resistance, pathogen resistance, blight resistance

原理：同义关系与标准术语

同义关系（Synonymy）

两个或多个不同的语言形式表达相同或相近的概念。术语映射的任务是识别这些变体，并将它们归并到**标准术语**（preferred term）。

- **完全同义**：`drought tolerance` ≡ `drought resistance`（在植物学语境下可互换）
- **部分同义/上下位**：`drought stress tolerance` ⊐ `drought tolerance`（前者更具体，后者更通用）
- **缩略/全称**：`OsDREB1A` = `Oryza sativa Dehydration-Responsive Element-Binding protein 1A`

- **基因符号变体**：`OsDREB1A` = `OsDREB1` = `LOC_Os01g0712800`（NCBI Gene ID）
- **蛋白质名称变体**：`OsDREB1A` = `DREB1A protein` = `dehydration-responsive element-binding protein 1A`

术语表的结构

表面形式（变体）	标准术语	术语ID（Trait Ontology）
drought tolerance	drought tolerance	TO:0006001
drought resistance	drought tolerance	TO:0006001
drought stress tolerance	drought tolerance	TO:0006001
water deficit tolerance	drought tolerance	TO:0006001
salt tolerance	salt tolerance	TO:0006002
salinity tolerance	salt tolerance	TO:0006002
salt stress tolerance	salt tolerance	TO:0006002
NaCl tolerance	salt tolerance	TO:0006002
cold tolerance	cold tolerance	TO:0006003
chilling tolerance	cold tolerance	TO:0006003
freezing tolerance	cold tolerance	TO:0006003

- **等价类（Equivalence Class）**：同一标准术语下的所有变体互为同义词，可互换使用。
- **层级关系（Hierarchy）**：术语表通常组织为层级结构，支持泛化查询。
  - `drought tolerance` □ `abiotic stress tolerance` □ `stress tolerance` □ `plant trait`
- **字符串匹配策略**：精确匹配→ 忽略大小写匹配→ 词形还原后匹配（与第一步协同）。

与词形还原的协同示例

```
# 第一步：词形还原
text = "drought-stress-tolerance"
doc = nlp(text)
lemma = "-".join([t.lemma_ for t in doc])
# "drought stress tolerance（名词，无变化）"

# 第二步：术语映射
TERMINOLOGY_MAP = {
    "drought-tolerance": "TO:0006001",
    "drought-resistance": "TO:0006001",
    "drought-stress-tolerance": "TO:0006001",
    "water-deficit-tolerance": "TO:0006001",
    "salt-tolerance": "TO:0006002",
    "salinity-tolerance": "TO:0006002",
    "salt-stress-tolerance": "TO:0006002",
    "cold-tolerance": "TO:0006003",
    "chilling-tolerance": "TO:0006003",
```

```
    }
    standard_id = TERMINOLOGY_MAP.get(lemma)
    # "drought stress tolerance" -> "TO:0006001"
```

局限：术语表的边界

术语映射依赖**人工构建的术语表**，存在天然局限：

- **覆盖不全**：新发现的基因（如 OsNAC6、OsWRKY45）、新命名的性状无法及时收录
- **领域局限**：apple 在植物学中是水果，在计算机领域是公司；resistance 在植物病理学中是抗病性，在电学中是电阻
- **语境依赖**：OsDREB1A 在分子生物学中是基因，在育种学中可能指代品种；Cold 在植物学中是低温胁迫，在医学中是感冒
- **层级模糊**：stress tolerance 是 drought tolerance 的上位词，但某些文献中两者混用

这些需要第三步实体链接，通过上下文消歧。

为什么需要实体链接？

以下句子中的 OsDREB1A 是同一个实体吗？

- OsDREB1A improves drought tolerance.（分子生物学语境：一个转录因子基因）
- OsDREB1A was released in 2010.（育种学语境：可能指代一个水稻品种）
- OsDREB1A was identified by QTL mapping.（遗传学语境：可能指代一个 QTL 位点）

更极端的例子：apple 可能是水果、公司、唱片公司；Washington 可能是人名、地名、大学名。仅凭表面形式无法确定，必须结合**上下文**进行消歧。

生物学中的同名异义实例

表面形式	可能含义1	可能含义2
OsDREB1A	基因（Gene）	蛋白质（Protein）
Cold	低温胁迫（Abiotic stress）	感冒（Disease）
Apple	水果（Fruit）	公司（Apple Inc.）
Virus	病毒（Pathogen）	计算机病毒（Software）
Cell	细胞（Biology）	电池（Battery）
Culture	组织培养（Tissue culture）	文化（Culture）

原理：命名实体与消歧

命名实体（Named Entity）

文本中具有特定指称的专有名词，如基因名、蛋白质名、疾病名、地名、机构名。命名实体通常有唯一标识符（ID），存储在知识库中。

- **实体提及（Entity Mention）**：文本中出现的实体表面形式，如 `OsDREB1A`、`OsNAC6`、`apple`。
- **知识库实体（KB Entity）**：知识库中的唯一条目，如 UniProt: Q0J8P9（OsDREB1A 蛋白质）、NCBI Gene: 4340976（OsNAC6 基因）。
- **实体链接任务**：建立实体提及→ 知识库实体的映射，需要解决**同名异义**（ambiguity）和**异名同义**（synonymy）。

生物学知识库示例

知识库	覆盖范围	示例ID
UniProt	蛋白质序列与功能	Q0J8P9（OsDREB1A）
NCBI Gene	基因信息	4340976（OsNAC6）
Trait Ontology (TO)	植物性状	TO:0006001（drought tolerance）
Gene Ontology (GO)	基因功能分类	GO:0009414（response to water deprivation）
Plant Ontology (PO)	植物解剖与发育	PO:0009005（root）

消歧原理：四步流程

1. **候选生成（Candidate Generation）**
   根据表面形式，从知识库中检索所有可能实体。如 `OsDREB1A` 的候选：UniProt: Q0J8P9（蛋白质）、NCBI Gene: 4340976（基因）、RAP-DB: Os01g0712800（位点）。
2. **上下文编码（Context Encoding）**
   将实体提及的上下文（整个句子或段落）编码为稠密向量（dense vector），捕捉语义信息。如 "OsDREB1A improves drought tolerance" 的向量编码。
3. **相似度计算（Similarity Computation）**
   计算上下文向量与每个候选实体描述向量的相似度。如：
   - "OsDREB1A improves drought tolerance" vs "Q0J8P9: transcription factor involved in drought response" → 高相似度
   - "OsDREB1A improves drought tolerance" vs "Q0J8P9: rice variety released in 2010" → 低相似度
4. **排序选择（Ranking & Selection）**
   选择相似度最高的候选实体作为最终链接结果。若最高相似度低于阈值，则标记为"未链接"（NIL）。

与主谓宾抽取的结合

```python
# 完整的知识条目标准化流程（四步）
def extract_svo_standardized(doc, entity_linker):
    # 第一步：词形还原
    # 第二步：术语映射（可选，用于性状）
    # 第三步：SVO 抽取（含从句处理）
    svo = extract_svo_v3(doc)
    if svo is None:
        return None

    # 第四步：实体链接（用于基因蛋白质等命名实体）/
    subj_kb = entity_linker.link(svo[0], context=doc.text)
    obj_kb  = entity_linker.link(svo[2], context=doc.text)

    return (subj_kb, svo[1], obj_kb)
    # ('UniProt:Q0J8P9', 'improve', 'TO:0006001')
    # ('NCBI:4340976', 'enhance', 'TO:0006002')  # OsNAC6 例子
```

三步策略的总结

步骤	解决的问题	依赖资源	生物学例句
词形还原	拼写不同，词根相同	词典、词形规则	improves → improve
术语映射	说法不同，概念相同	领域术语表	drought resistance → TO:0006001
实体链接	同名异义，异名同义	大规模知识库、消歧模型	OsDREB1A → UniProt:Q0J8P9

生物学实体标准化实例

输入句子	标准化结果	说明
`OsDREB1A improves drought tolerance.`	`(Q0J8P9, improve, TO:0006001)`	基因-性状关系
`OsNAC6 enhances salt tolerance.`	`(NCBI:4340976, enhance, TO:0006002)`	基因-性状
`OsWRKY45 binds to W-box element.`	`(Q7XSQ6, bind, GO:0003677)`	蛋白质-DNA 结合
`SNAC1 regulates stomatal closure.`	`(Q6Z4I5, regulate, GO:0090332)`	蛋白质-过程

实际应用中的简化

在植物基因-性状关系抽取的实际项目中，通常采用**混合策略**：

- 基因/蛋白质：实体链接（对接 UniProt、NCBI Gene）
- 性状：术语映射（对接 Trait Ontology、Plant Trait Ontology）
- 动词：词形还原（无需链接，动作类型有限）

这样平衡了精度与计算成本。