PubMed 中的生物实体识别（NER）、Shell 编程与 NLTK 实现

Jingbo Xia

华中农业大学（Huazhong Agricultural University）

xiajingbo.math@gmail.com

2026-10-08

PubTator 是由 NCBI 团队持续维护的 PubMed 生物实体标注数据库，经由 PubTator 的自动注释，PubMed 中的摘要数据和 PMC 中的全文数据均获得了较为广泛的实体标注

——主要覆盖 Gene, Chemical, Species, Mutation, Disease 等几个常用类别 [a] 。

获取海量的 PubTator 标注文件并非难事，知识的进一步挖掘和加工，是主要的考验。让我们来尝试进行相关的训练吧。

[a] 参见https://www.ncbi.nlm.nih.gov/research/pubtator/。

本讲的工具地图

共性

- 都是“文本 → 结构化信息”的转换器；
- 输出均可复核：字符偏移 / 句子编号；
- 都可编程访问：RESTful API / Python 库；
- 都不直接产出生物学结论。

区别

	PubTator	NLTK
定位	生物文献标注服务与数据库	通用 NLP 教学库
知识	内置 BioNER 模型 + 标准标识符	无领域知识
输出	实体提及 + 标准编号	句子、词元、词频
运行	远程 API，随模型版本变化	本地运行，版本可控
分工	提供标注（原材料）	负责加工（结构化）

本讲的叙事线

PubTator 获取标注 → Shell 批量下载 → RiceMind 对照（已结构化证据）→ NLTK 加工为句子级证据表。

Table of contents I

1. PubMed 5
2. PubTator 24
3. RiceMind 数据与 API 访问 32
4. 代码实验：NLTK 实现文本预处理 38
   - 数据结构与环境准备 39
   - NLTK 基础实现 45
   - 句子级证据构建 50
   - 验证与课堂练习 54
5. 讨论：面向水稻数据的自主实体识别开发 57

1 PubMed 5

2 PubTator 24

3 RiceMind 数据与 API 访问 32

4 代码实验：NLTK 实现文本预处理 38
- 数据结构与环境准备 39
- NLTK 基础实现 45
- 句子级证据构建 50
- 验证与课堂练习 54

5 讨论：面向水稻数据的自主实体识别开发 57



PubMed® comprises more than 33 million citations for biomedical literature from MEDLINE, life science journals, and online books. Citations may include links to full-text content from PubMed Central and publisher websites.

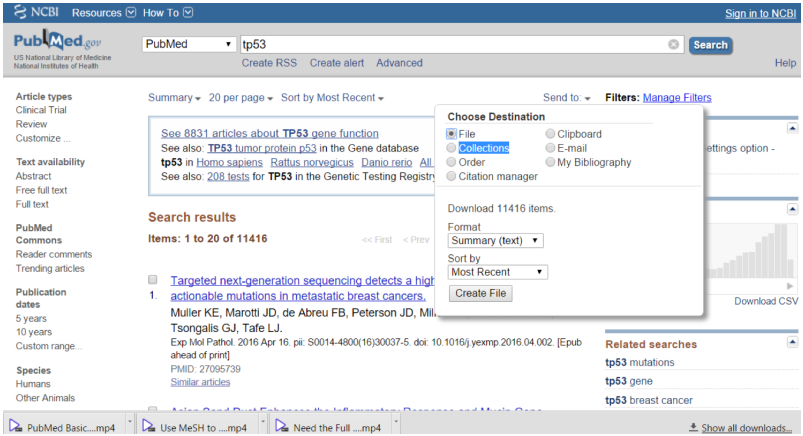


图 1: Previous version



图 2: PubMed:https://pubmed.ncbi.nlm.nih.gov/

- Video: "Welcome to E-utilities for PubMed" (February 13, 2018)
  https://www.nlm.nih.gov/oet/ed/insidersguide/welcome-to-e-utilities.html
- Page: "How does E-utilities work?"
  https://dataguide.nlm.nih.gov/eutilities/how_eutilities_works.html

Example links:
https://eutils.ncbi.nlm.nih.gov/entrez/eutils/esearch.fcgi?db=pubmed&term=nature[journal]+AND+3D+printing

This XML file does not appear to have any style information associated with it. The document tree is shown below.


```
▼<eSearchResult>
  <Count>51</Count>
  <RetMax>20</RetMax>
  <RetStart>0</RetStart>
  ▼<IdList>
    <Id>34408301</Id>
    <Id>34194019</Id>
    <Id>33790446</Id>
    <Id>33692557</Id>
```


课堂思考

默认下载 20 个 pmid。

```
<TermSet>
  <Term>"printing, three-dimensional"[MeSH Terms]</Term>
  <Field>MeSH Terms</Field>
  <Count>9433</Count>
  <Explode>Y</Explode>
</TermSet>
<TermSet>
  <Term>"printing"[All Fields]</Term>
  <Field>All Fields</Field>
  <Count>30858</Count>
  <Explode>N</Explode>
</TermSet>
<TermSet>
  <Term>"three-dimensional"[All Fields]</Term>
  <Field>All Fields</Field>
  <Count>270008</Count>
  <Explode>N</Explode>
```

课堂思考

Translation 项有益于指导文献细分。

https://eutils.ncbi.nlm.nih.gov/entrez/eutils/esearch.fcgi?db=pubmed&term=nature[journal]+AND+3D+printing&retmax=50&usehistory=y

This XML file does not appear to have any style information associated with it. The document tree is shown below.


```
▼<eSearchResult>
   <Count>51</Count>
   <RetMax>50</RetMax>
   <RetStart>0</RetStart>
   <QueryKey>1</QueryKey>
   <WebEnv>MCID_6249b847ad450b5f43183060</WebEnv>
  ▼<IdList>
     <Id>34408301</Id>
     <Id>34194019</Id>
     <Id>33790446</Id>
     <Id>33692557</Id>
     <Id>33597754</Id>
     <Id>33361791</Id>
     <Id>33361786</Id>
     <Id>33204016</Id>
     <Id>32332909</Id>
```

课堂思考

usehistory=y 用于获取 WebEnv。

https://eutils.ncbi.nlm.nih.gov/entrez/eutils/efetch.fcgi?db=pubmed&id=27558065,27466125,27127816,26842040&rettype=abstract&retmode=text

```
1. Nature. 2016 Aug 25;536(7617):451-5. doi: 10.1038/nature19100.

An integrated design and fabrication strategy for entirely soft, autonomous
robots.

Wehner M(1)(2), Truby RL(1)(2), Fitzgerald DJ(1)(2), Mosadegh B(3)(4), Whitesides
GM(2)(5), Lewis JA(1)(2), Wood RJ(1)(2).

Author information:
(1)John A. Paulson School of Engineering and Applied Sciences, Harvard
University, Cambridge, Massachusetts 02138, USA.
(2)Wyss Institute for Biologically Inspired Engineering, Harvard University,
Cambridge, Massachusetts 02138, USA.
(3)Dalio Institute of Cardiovascular Imaging, Weill Cornell Medicine and New York
Presbyterian Hospital, New York, New York 10021, USA.
(4)Department of Radiology, Weill Cornell Medicine, New York, New York 10021,
USA.
(5)Department of Chemistry and Chemical Biology, Harvard University, Cambridge,
Massachusetts 02138, USA.

Comment in
    Nature. 2016 Aug 25;536(7617):400-1.

Soft robots possess many attributes that are difficult, if not impossible, to
achieve with conventional robots composed of rigid materials. Yet, despite recent
advances, soft robots must still be tethered to hard robotic control systems and
power sources. New strategies for creating completely soft robots, including soft
analogues of these crucial components, are needed to realize their full
potential. Here we report the untethered operation of a robot composed solely of
```

课堂思考

下载指定 pmid 的文献

课堂思考

如果要下载 50 篇有关 Covid19 的文献？

API: https://eutils.ncbi.nlm.nih.gov/entrez/eutils/esearch.fcgi?db=pubmed&term=covid19[mesh]&retmax=50&usehistory=y

https://eutils.ncbi.nlm.nih.gov/entrez/eutils/esearch.fcgi?db=pubmed&term=covid19[mesh]&retmax=50&usehistory=y

This XML file does not appear to have any style information asso


```
▼<eSearchResult>
   <Count>150159</Count>
   <RetMax>50</RetMax>
   <RetStart>0</RetStart>
   <QueryKey>1</QueryKey>
   <WebEnv>MCID_6249b44e27e59437ea0f9898</WebEnv>
  ▼<IdList>
     <Id>35356847</Id>
     <Id>35356758</Id>
     <Id>35356660</Id>
     <Id>35356652</Id>
     <Id>35356648</Id>
```

API: https://eutils.ncbi.nlm.nih.gov/entrez/eutils/efetch.fcgi?db=pubmed&query_key=1&WebEnv=MCID_6249b44e27e59437ea0f9898&retstart=0&retmax=50&retmode=text&rettype=abstract

$ curl "https://eutils.ncbi.nlm.nih.gov/entrez/eutils/efetch.fcgi?db=pubmed&query_key=1&WebEnv=MCID_6249b44e27e59437ea0f9898&retstart=0&retmax=50&retmode=text&rettype=abstract"

Done!

课堂思考

如果不是下载 50 篇，而要下载所有有关 Covid19 的文献？

retmax=???

课堂思考

如果要获得所有有关 Covid19 的文献的 pmid 列表？

API: https://eutils.ncbi.nlm.nih.gov/entrez/eutils/esearch.fcgi?db=pubmed&term=covid19[mesh]&retmax=150159

Note: 9999 restriction applied. （150159 为当时查询的示例数值，实际结果数随时间变化。）

$ curl "https://eutils.ncbi.nlm.nih.gov/entrez/eutils/esearch.fcgi?db=pubmed&term=covid19[mesh]&retmax=150159" >pmid_covid19.txt

This XML file does not appear to have any style information associated with it. The document tree is shown below.

```
▼<eSearchResult>
   <Count>150159</Count>
   <RetMax>150159</RetMax>
   <RetStart>0</RetStart>
  ▼<IdList>
     <Id>35356847</Id>
     <Id>35356758</Id>
     <Id>35356660</Id>
     <Id>35356652</Id>
     <Id>35356648</Id>
     <Id>35356314</Id>
     <Id>35356264</Id>
     <Id>35356089</Id>
     <Id>35356088</Id>
     <Id>35356029</Id>
     <Id>35356028</Id>
     <Id>35356027</Id>
     <Id>35356026</Id>
     <Id>35356025</Id>
     <Id>35356024</Id>
     <Id>35356023</Id>
```

cat pmid_covid19.txt | grep "<Id>"

```
cat pmid_covid19.txt | grep "<Id>" | sed 's/<Id>//g'
cat pmid_covid19.txt | grep "<Id>" | sed 's/<Id>//g' | sed 's/<\/Id>//g'
cat pmid_covid19.txt | grep "<Id>" | sed 's/<Id>//g' | sed 's/<\/Id>//g' | sed 's/[[:blank:]]//g'
```

课堂思考

如果要依据 pmid 获得论文摘要呢？

API: https://eutils.ncbi.nlm.nih.gov/entrez/eutils/efetch.fcgi?db=pubmed&id=39371582&retmode=xml

1 PubMed 5

2 PubTator 24

3 RiceMind 数据与 API 访问 32

4 代码实验：NLTK 实现文本预处理 38
- 数据结构与环境准备 39
- NLTK 基础实现 45
- 句子级证据构建 50
- 验证与课堂练习 54

5 讨论：面向水稻数据的自主实体识别开发 57



图 3: PubTator interface

PubTator [1] is a Web-based tool for accelerating manual literature curation (e.g. annotating biological entities and their relationships) through the use of advanced text-mining techniques. As an all-in-one system, PubTator provides one-stop service for annotating PubMed citations.

[1] https://www.ncbi.nlm.nih.gov/research/pubtator/api.html

Access annotations via RESTful API

PubTator users can programmatically access our computer annotation results through RESTful API.

API format (Pubtator, JSON, BioC) [a]

Syntax:

`https://www.ncbi.nlm.nih.gov/research/pubtator-api/publications/export/[Format]?[Type]=[Identifiers]`

`&concepts=[Bioconcepts]`

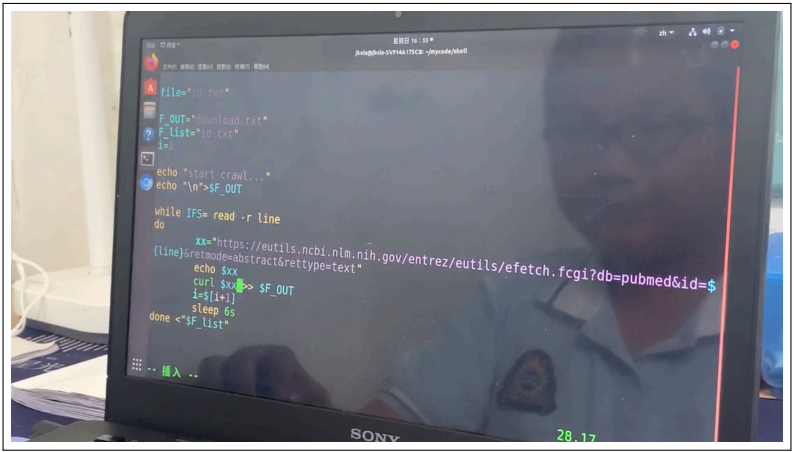
[a] https://www.ncbi.nlm.nih.gov/research/pubtator/api.html.

For example:

https://www.ncbi.nlm.nih.gov/research/pubtator-api/publications/export/pubtator?pmids=19894120

```
19894120|t|Lipopolysaccharide increases the expression of multidrug resistance-associated protein 1 (MRP1) in RAW 264.7 macrophages.
19894120|a|Multidrug resistance-associated protein 1 (MRP-1) is a ubiquitously expressed member of the ATP-binding cassette transporter family.
MRP-1 is one of the primary transporters of glutathione and glutathione conjugates. This protein also transports antiretroviral therapeutics,
such as HIV-1 protease inhibitors (PI). We hypothesized that inflammatory mediators that activate macrophages would modify the expression and
activity of MRP-1 in macrophages. Real-time PCR assays, western blots, and calcein efflux assays were used to show that exposure of macrophage
cell line RAW 264.7 to lipopolysaccharide (LPS) increased expression of MRP-1 at the levels of mRNA, protein, and functional activity. Treatment
of macrophages with LPS resulted in 2-fold increases of MRP-1 expression or functional activity. LPS-mediated increases in calcein efflux were
repressed by the MRP-specific inhibitor MK-571. These results suggest that the effectiveness of HIV-1 PI therapy may be compromised by the
presence of opportunistic infections.
19894120        0       18      Lipopolysaccharide      Chemical        MESH:D008070
19894120        47      88      multidrug resistance-associated protein 1       Gene    4363
19894120        90      94      MRP1    Gene    4363
19894120        99      108     RAW 264.7       CellLine        CVCL_0493;NCBITaxID:10090
19894120        122     163     Multidrug resistance-associated protein 1       Gene    4363
19894120        165     170     MRP-1   Gene    4363
19894120        255     260     MRP-1   Gene    4363
19894120        299     310     glutathione     Chemical        MESH:D005978
19894120        315     326     glutathione     Chemical        MESH:D005978
19894120        405     410     HIV-1   Species 11676
19894120        551     556     MRP-1   Gene    4363
```

https://www.ncbi.nlm.nih.gov/research/pubtator-api/publications/export/biocxml?pmids=19894120&concepts=gene


file="id.txt"
F_OUT="download.txt"
F_list="id.txt"
i=1
echo "start crawl..."
echo "\n">$F_OUT
while IFS= read -r line
do
xx="https://eutils.ncbi.nlm.nih.gov/entrez/eutils/efetch.fcgi?db=pubmed&id=${line}&retmode=abstract&rettype=text"
echo $xx
curl $xx >> $F_OUT
i=$[i+1]
sleep 6s
done <"$F_list"
-- 插入 --
SONY

Shell programming

```
#!/bin/bash
F_OUT="tmp.txt"

echo -e "\n Let us curate the results!\n"

curl https://www.ncbi.nlm.nih.gov/research/pubtator-api/publications/export/pubtator?pmids=19894122 >$F_OUT
```

$ chmod 755 NCBI1.sh
if you do not have permission to run the shell.

Note：
定义一个名为 F_OUT 的变量，其值为"tmp.txt"
echo 命令：用于在终端输出一行文本。-e 选项：启用反斜杠转义符的解释。有了这个选项，\n 才会被识别为换行符，而不是普通的字符。

Shell programming

```
#!/bin/bash
F_OUT="result_Pubtator.txt"
F_LIST="pmid_list.txt"

echo -e "\n Let us curate the results!\n"
echo -e "\n" >$F_OUT

total=$(wc -l < "$F_LIST")
i=1
while IFS= read -r line
do
 printf "%d-th result out of %d is processing...\n" "$i" "$total"
 i=$((i+1))
done <"$F_LIST"
```

Note:
$ 变量名用于获取变量的值

Shell programming

```
#!/bin/bash
F_OUT="result_Pubtator.txt"
F_LIST="pmid_list.txt"

echo -e "\n Let us curate the results!\n"
echo -e "\n" >$F_OUT

total=$(wc -l < "$F_LIST")
i=1
while IFS= read -r line
do
 url="https://www.ncbi.nlm.nih.gov/research/pubtator-api/publications/export/pubtator?pmids=$line"
 curl "$url" >>$F_OUT
 printf "%d-th result out of %d is processing...\n" "$i" "$total"
 i=$((i+1))
 sleep 5.8s
done <"$F_LIST"
```



1 PubMed 5

2 PubTator 24

3 RiceMind 数据与 API 访问 32

4 代码实验：NLTK 实现文本预处理 38
- 数据结构与环境准备 39
- NLTK 基础实现 45
- 句子级证据构建 50
- 验证与课堂练习 54

5 讨论：面向水稻数据的自主实体识别开发 57

RiceMind 面向水稻文献证据的查询与整理，将基因、性状、品种、PMID 以及句子证据组织为可追溯的数据对象。本章的目标不是“看一个数据库页面”，而是学会提出查询、读取 JSON、保留证据来源，并用 Shell 完成可复现筛选。

从文献到可计算证据

- 文献层：PMID、题名、期刊、年份、DOI；
- 证据层：句子文本（`text`）及其句子编号（`sent_id`）；
- 实体层：基因（Gene）、性状（Trait）和品种（Variety）；
- 评价层：来源（`sources`）、证据代码和置信度层级。

课堂思考

如果一个基因在很多句子中出现，是否就可以直接断言它造成了某个性状？为什么还要保留原始句子和证据代码？

RiceMind API

基础地址

http://lit-evi.hzau.edu.cn/ricemind-api/

按基因查询

Gene profile:

http://lit-evi.hzau.edu.cn/ricemind-api/gene-profile/?gene=sd1

基因相关的句子证据:

http://lit-evi.hzau.edu.cn/ricemind-api/search-by-gene/?gene=sd1&page=1&limit=20

基因关联的性状:

http://lit-evi.hzau.edu.cn/ricemind-api/traits-by-gene/?gene=sd1&confidence=All&onto_type=ALL&page=1&limit=20

按性状及基因组合查询

按性状查询

RiceMind 中的性状名称应使用数据库中保存的标准标签。例如：

`drought tolerance`

http://lit-evi.hzau.edu.cn/ricemind-api/search-by-trait/?trait=drought%20tolerance&page=1&limit=20

按基因–性状组合查询

http://lit-evi.hzau.edu.cn/ricemind-api/search-by-trait-and-gene/?gene=sd1&trait=drought%20tolerance&page=1&limit=20

解释边界

API 返回的候选记录、置信度和证据代码用于组织与筛选文献证据，不等同于最终的生物学结论；句子中的共现关系也不自动等于因果关系。

读取 **JSON** 时应保留什么？

- 分页信息：current_page、total_pages、total_count；
- 文献定位：PMID、year、journal、title、doi；
- 句子定位：sent_id、text；
- 证据解释：gene、trait、confidence_tier、evidence_codes；
- 可追溯性：原始 JSON 文件、查询参数和下载日期。

课堂思考

为什么不能只保存“基因名 + 性状名”的二维表？请从复核证据、去除误匹配和复现实验三个角度讨论。

```bash
#!/bin/bash
BASE='http://lit-evi.hzau.edu.cn/ricemind-api'
GENE='sd1'

# 保存一个基因的概览信息
curl -fsSL "$BASE/gene-profile/?gene=$GENE" \
  -o "$GENE.profile.json"

# 保存该基因的第1页句子证据
curl -fsSL "$BASE/search-by-gene/?gene=$GENE&page=1&limit=20" \
  -o "$GENE.evidence.page1.json"
```

检查与讨论

- 检查返回结果是否为合法 JSON，找出 `total_pages` 并思考如何循环下载全部页面；
- 保存原始响应并记录查询参数（基因、性状、`page`、`limit`），保证结果可复现。

课堂思考

能否仿照前面 PubTator 的代码：自动遍历所有页面，筛选包含目标性状的句子，并同时输出 PMID、sent_id、text 和原始 JSON？

1 PubMed 5

2 PubTator 24

3 RiceMind 数据与 API 访问 32

4 代码实验：NLTK 实现文本预处理 38
- 数据结构与环境准备 39
- NLTK 基础实现 45
- 句子级证据构建 50
- 验证与课堂练习 54

5 讨论：面向水稻数据的自主实体识别开发 57

1 PubMed 5

2 PubTator 24

3 RiceMind 数据与 API 访问 32

4 代码实验：NLTK 实现文本预处理 38
- 数据结构与环境准备 39
- NLTK 基础实现 45
- 句子级证据构建 50
- 验证与课堂练习 54

5 讨论：面向水稻数据的自主实体识别开发 57

本章的基本问题

如何把带有生物医学标注的文本，转换成可以检索、统计和复核的句子级证据？

PubTator 文本 ⟶ NLTK 预处理 ⟶ 句子级证据表

回顾开头的工具地图：PubTator 提供标注，NLTK 负责加工；本节把这条分工落到代码上。

INPUT
一篇或多篇摘要；
原始文本；
实体标注及其字符位置；
关键词或实体类型。

OUTPUT
句子编号；
句子文本；
实体提及及其类型；
PMID 和标准化标识符。

课堂思考

每一步都要回答三个问题：输入是什么？代码改变了什么？输出能否被别人复核？

```
20301443|t|Peutz-Jeghers syndrome
20301443|a|STK11 mutations cause this syndrome.

20301443<TAB>0<TAB>22<TAB>
Peutz-Jeghers syndrome<TAB>
Disease<TAB>MESH:D010580
20301443<TAB>23<TAB>28<TAB>
STK11<TAB>Gene<TAB>6794
```

<TAB> 表示原文件中的制表符。偏移量基于“标题 + 空格 + 摘要”的拼接文本，从 0 开始计数，end 为开区间。请验证：`text[23:28]` 是否等于`"STK11"`。

字段含义
- `|t|`：标题；
- `|a|`：摘要；
- `begin/end`：字符偏移；
- `mention`：文本中的提及；
- `type`：实体类型；
- `identifier`：标准编号。

教学重点

NLTK 负责文本结构处理；PubTator 负责提供生物医学实体及其标准化标识。

```
from pathlib import Path

def parse_pubtator(path):
    pmid, title, abstract, entities = None, "", "", []
    for line in Path(path).read_text(encoding="utf-8").splitlines():
        if "|t|" in line:
            pmid, title = line.split("|t|", 1)
        elif "|a|" in line:
            pmid, abstract = line.split("|a|", 1)
        elif line.strip():
            f = line.split("\t")
            entities.append({"pmid": f[0], "begin": int(f[1]),
                             "end": int(f[2]), "mention": f[3],
                             "type": f[4], "identifier": f[5]})
    return {"text": title + " " + abstract, "entities": entities}
```

关键约定

返回的 `text` 使用与偏移量一致的拼接规则（标题 + 空格 + 摘要），因此每个实体都应满足 `text[begin:end] == mention`，这本身就是一条自动检查。

适合完成

- 句子切分；
- 词元化；
- 词频统计；
- 基础文本清理；
- 句子级结果组织。

不能过度承诺

- NLTK 不是生物医学知识库；
- 通用 NER 不等于 BioNER；
- 词频不等于生物学重要性；
- 自动切分结果需要抽样检查。

本章的原则

让工具处理重复性文本操作，让研究者保留数据定义、结果检查和科学解释。

```
pip install nltk
```

```
import nltk
nltk.download("punkt")      # sent_tokenize / word_tokenize 所需
nltk.download("punkt_tab")  # NLTK >= 3.9 还需要此资源
```

常见报错

若运行 `sent_tokenize` 时报 `LookupError: Resource punkt not found`，说明上述数据包尚未下载；这是本章实验最常见的环境问题。

课堂思考

为什么 NLTK 把“代码”和“语言模型数据”分开安装？这对结果的可复现性意味着什么？

1 PubMed 5

2 PubTator 24

3 RiceMind 数据与 API 访问 32

4 代码实验：NLTK 实现文本预处理 38
- 数据结构与环境准备 39
- NLTK 基础实现 45
- 句子级证据构建 50
- 验证与课堂练习 54

5 讨论：面向水稻数据的自主实体识别开发 57

INPUT `abstract.txt`，UTF-8 编码的一篇摘要。

```
from pathlib import Path
from nltk.tokenize import sent_tokenize

text = Path("abstract.txt").read_text(encoding="utf-8")
sentences = sent_tokenize(text)

for sent_id, sentence in enumerate(sentences, start=1):
    print(sent_id, sentence)
```

OUTPUT 的含义

输出不是“若干行字符串”而已，而是一个有顺序的句子列表；`sent_id` 将成为后续证据定位的索引。

INPUT

```
P53 mutations were detected in
several samples.
The mutations were associated with
tumor progression.
```

OUTPUT

```
sentence_id | sentence
1 | P53 mutations were detected...
2 | The mutations were
associated...
```

学生应能解释

`sent_tokenize` 把一个长字符串转换成句子序列；它没有识别 Gene 或 Disease，只完成文本结构切分。

（以上为结构示意输出，不是完整终端截图。）

INPUT 上一案例得到的句子列表。

```
from collections import Counter
from nltk.tokenize import word_tokenize

tokens = []
for sentence in sentences:
    tokens.extend(word_tokenize(sentence))

words = [
    token.lower()
    for token in tokens
    if token.isalpha()
]

frequency = Counter(words)
for word, count in frequency.most_common(10):
    print(word, count)
```

处理链

sentence ⟶ tokens ⟶ 小写化 ⟶ 去除标点 ⟶ 词频表

中间结果：词元

```
["P53", "mutations", "were",
"detected", "in", "samples"]
```

清理后

```
["mutations", "were",
"detected", "samples"]
```

OUTPUT：词频表

```
word        frequency
mutations      12
detected        9
samples         7
```

解释边界

词频高不等于生物学重要；数字、基因符号、连字符和大小写处理都会改变统计结果。

1 PubMed 5

2 PubTator 24

3 RiceMind 数据与 API 访问 32

4 代码实验：NLTK 实现文本预处理 38
- 数据结构与环境准备 39
- NLTK 基础实现 45
- 句子级证据构建 50
- 验证与课堂练习 54

5 讨论：面向水稻数据的自主实体识别开发 57

```python
from nltk.tokenize import sent_tokenize

def sentence_spans(text):
    spans, cursor = [], 0
    for sent_id, sent in enumerate(sent_tokenize(text), start=1):
        start = text.find(sent, cursor)
        if start == -1:
            raise ValueError("sentence alignment failed")
        end = start + len(sent)
        spans.append({"sent_id": sent_id, "begin": start,
                      "end": end, "text": sent})
        cursor = end
    return spans
```

为什么要保留字符位置？

PubTator 实体标注使用字符偏移；先定位句子的 `begin/end`，才能判断实体属于哪个句子。

这段代码的脆弱性

`text.find` 要求切分结果与原文逐字符一致；空白字符被改动则对齐失败。

PMID	sentence_id	mention	type	identifier
20301443	1	Peutz-Jeghers syndrome	Disease	MESH:D010580
20301443	1	STK11	Gene	6794

OUTPUT 记录的含义
每一行都能回到一篇文献、一个句子和一个实体；原始文本、字符位置和标准编号共同构成可复核的证据。

从文本处理到知识发现

NLTK 并没有直接得出“STK11 导致某疾病”的结论；它只是帮助我们建立稳定、可检查的证据单元。

课堂思考

如果一个实体跨越两个句子，或者句子切分发生错误，应该如何标记、复核和修正？

对比：PubTator 需要字符偏移对齐；RiceMind 已给出句子级证据（`sent_id`、`text`），NLTK 的任务从“对齐”变为“筛选与统计”。

```
import json
from pathlib import Path
from nltk.tokenize import word_tokenize

hits = []
for path in sorted(Path(".").glob("sd1.evidence.page*.json")):
    data = json.loads(path.read_text(encoding="utf-8"))
    for item in data["results"]:  # 键名以API实际返回为准
        text = item["text"]
        words = [w.lower() for w in word_tokenize(text)]
        if "drought" in words:
            hits.append((item["PMID"], item["sent_id"], text))

for pmid, sent_id, text in hits:
    print(pmid, sent_id, text, sep="\t")
```

与 PubTator 流程的分工差异

PubTator 提供原始标注，NLTK 负责建立句子证据单元；RiceMind 提供已结构化的句子证据，NLTK 负责词级筛选与统计。两者最终都输出“PMID + `sent_id` + `text`”的可复核记录。

1 PubMed 5

2 PubTator 24

3 RiceMind 数据与 API 访问 32

4 代码实验：NLTK 实现文本预处理 38
- 数据结构与环境准备 39
- NLTK 基础实现 45
- 句子级证据构建 50
- 验证与课堂练习 54

5 讨论：面向水稻数据的自主实体识别开发 57

CHECK

- 句子数量是否合理？
- 每个实体是否仍在原句中？
- `begin/end` 是否对应正确字符？
- PMID 和实体编号是否保留？
- 抽样阅读结果是否可信？

LIMITATION

- 缩写和小数可能造成错误切分；
- 基因符号可能被当作普通词；
- 连字符和括号会影响词元化；
- 通用 NLTK 模型不等于专业 BioNER。

本章结论

可复现的数据分析不仅需要代码，还需要明确的输入、结构化的输出和可解释的验证步骤。

给定
一批 PubTator 摘要、一个关键词集合和实体类型集合。

请完成

- 使用 NLTK 切分句子；
- 为每个句子分配 `sent_id`；
- 找出包含目标关键词的句子；
- 输出 PMID、句子编号、句子文本、实体提及、实体类型和标准编号；
- 随机抽取若干结果，检查句子边界和实体对齐。

课堂思考

提交结果时，不只提交代码；还要提交一条输入样例、一条输出样例和一条你发现的错误或局限。

5. 讨论：面向水稻数据的自主实体识别开发

Outline I

1. PubMed 5
2. PubTator 24
3. RiceMind 数据与 API 访问 32
4. 代码实验：NLTK 实现文本预处理 38
   - 数据结构与环境准备 39
   - NLTK 基础实现 45
   - 句子级证据构建 50
   - 验证与课堂练习 54
5. 讨论：面向水稻数据的自主实体识别开发 57

PubTator 覆盖的实体类型
gene、disease、chemical、species、mutation、cell line
——为人类医学文献训练的模型。

RiceMind 需要的实体
作物性状条目（grain yield、drought tolerance、tillering……，对应 TO/CO 本体）、品种名、QTL 位点等
——PubTator 完全没有这一类。

回到开头的工具地图

当 API 的领域知识不够用时，缺口要由自己的代码补上：从 **API** 的使用者变成工具的构建者。

5. 讨论：面向水稻数据的自主实体识别开发

讨论

路线	做法	优点	代价
A. 词典匹配	RiceMind 拉性状词表建词典；NLTK 分词 +n-gram 匹配；对齐到句子	零训练数据、可解释、可复现	同义词覆盖不全
B. 共现拼接	PubTator 抽基因，词典抽性状，句子级共现	各取所长，改动最小	共现 $\neq$关联，需过滤
C. 弱监督扩展	用 RiceMind 句子作种子，提取上下文模式，迭代扩词表	能发现词典外的新性状词	工程量大，超出 NLTK 舒适区

路线 A 与 B 的零件全部是本课已讲过的：API 下载、NLTK 分词、句子对齐、证据表。路线 C 作为讨论方向，点到为止。

输入
RiceMind 性状词表 + PubMed 摘要文本。

示例逻辑

- 从 RiceMind 取性状词表：`["grain yield", "drought tolerance", ...]`；
- NLTK 对摘要做句切分与分词，生成 1–3 元 n-gram；
- n-gram 与词典（小写化后）匹配，命中即得性状提及；
- 复用 `sentence_spans` 把提及对齐到 `sent_id`，输出证据行。

示例输出

```
PMID + sent_id + "grain yield" + Trait + TO:0000397
```

输入
同一批 PubMed 摘要，PubTator API 在线标注。

示例逻辑

- PubTator 照常抽取基因提及（自带 GeneID）；
- 路线 A 的词典抽取同一文本中的性状提及；
- 两者都通过句子 span 对齐到 `sent_id`；
- 同一句内基因与性状共现 → 候选“基因–性状”对。

边界

共现不等于关联：“OsX increases grain yield”与“unlike OsX, grain yield ……”在共现层面无法区分，需要规则或人工过滤。

评估设计

- RiceMind 的策展证据（PMID+ 句子 + 基因 + 性状）作为金标准；
- 自建 pipeline 的输出与之对齐，计算 precision/recall；
- 错误分析：漏同义词 →recall 低；“yield”歧义 →precision 低。

课堂思考

开放讨论：词典法的天花板在哪里？换成玉米/小麦数据，哪些代码可以直接复用？什么情况下应该放弃 NLTK 规则、转向训练模型？

本节结论

PubTator、Shell、RiceMind、NLTK 四条线在此汇合：API 提供标注与金标准，NLTK 补上领域缺口，评估驱动迭代。

Acknowledgement

