哈爾濱工業大學
HARBIN INSTITUTE OF TECHNOLOGY

自然语言处理
第六章 词法分析

徐永东

ydxu@hit.edu.cn

词法分析

- 概述
- 英文的词法分析
  - 英文词识别
  - 英文词形还原
- 中文的词法分析
  - 形态分析
  - 分词
- 中文未登录词识别
  - 命名实体识别
  - 术语、新词

概述

词：是自然语言中能够独立运用的最小单位，是语言信息处理的基本单位。

词法分析：词汇层的分析技术

词的识别：将句子转换成词序列。

形态分析：词的构成、形态变化、词形还原。

词性标注：标记句子中的词的词性。

英文的词法分析

英文的词法分析

屈折型语言：

- 词之间一般有边界标记
- 词的形态变化丰富。

词法分析

- 词的识别(Tokenization)
  - 将句子转换成词序列。
  - 例子：I’m a student → I /’m/ a/ student/
- 词形还原(Lemmatization)
  - 分析词的形态结构：词的原型+形态变化。
  - 例子：takes → take + ~s；took → take + ~ed
- 词性标注：POS (Part-of-Speech) Tagging

英文词的识别——Tokenization

数字：123,456.78 90.7% 3/8 11/20/2000

缩略（包含不同的情况）：

字母－点号－字母－点号组成的序列，比如：U.S. 、i.e. 等；

字母开头，最后以点号结束，比如：Mr. Prof.、Dr. ；

包含非字母字符，比如：AT&T Micro$oft

带杠的词串，如：three-year-old，one-third，so-called

带瞥号的词串，如：I'm can't dog's let's

带空格的词串，如："and so on"，"ad hoc“

其他：如网址（http://insun.hit.edu.cn ）、公式等

英文词的识别——Tokenization

常见的特殊形式的英文词识别

Prof.、Mr.、 Ms 、 Co.、 Oct. 等放入词典；

Let’s 、 let’s => let + us

I’am => I + am

{it, that, this, there, what, where}’s =>

{it, that, this, there, what, where} + is

can’t => can + not; won’t => will + not

英文词的识别——Tokenization

常见的特殊形式的英文词识别

{is, was, are, were, has, have, had}n't =>

{is, was, are, were, has, have, had} + not

X've => X + have;

X'll => X + will;

X're => X + are

he's => he + is / has => ?

she's => she + is / has => ?

X'd Y => X + would (如果Y 为单词原型)=>

X + had (如果Y 为过去分词)

Tokenization问题

例外较多，跟文本来源有关

歧义现象（如点号的句子边界歧义）

数字的识别

数词的识别一般可以用有限状态自动机来实现

识别分数的正则表达式：

[0-9]+ / [0-9]+

例子：12/21

识别百分数的正则表达式：

([+ | -]) ? [0-9]+ (. [0-9]*) ? %

例子： -5.9%, 91%

识别十进制数字的正则表达式：

([0-9]+(,)?)+ (. [0-9]+)?

例子： 12,345

Tokenization code

```
import nltk
#nltk.download('punkt')
from nltk.tokenize import WordPunctTokenizer
import numpy as np
text= "Hello Mr. wang, welcome readers. I hope you hadn't find it 95% interesting . please do reply."
paragraph = text.lower()
sen_tokenizer = nltk.data.load('tokenizers/punkt/english.pickle')
sentences = sen_tokenizer.tokenize(paragraph)
for sen in sentences:
        nltk.word_tokenize(sen)
```

英文词形还原——Lemmatization

屈折型语言的词语变化形式：

屈折变化：即由于单词在句子中所起的语法作用的不同而发生的词的形态变化，而单词的词性基本不变的现象，如（take, took, takes）。识别这种变化是词法分析的最基本的任务。

派生变化：即一个单词从另外一个不同类单词或词干衍生过来，如morphological morphology，英语中派生变化主要通过加前缀或后缀的形式构成；在其他语言中，如德语和俄语中，同时还伴有音的变化。

复合变化：两个或更多个单词以一定的方式合成一个新的单词。这种变化形式比较灵活，如well-formed, 6-year-old等等。

Lemmatization的目的：将上述变化还原

英文词形还原——Lemmatization

规则变化的词形还原

ed 结尾的动词过去时，去掉**ed**;

ed → (e.g., worked → work)

*ed → *e (e.g., believed → believe)

*ied → *y (e.g., studied → study)

ing 结尾的现在分词，

*ing → * (e.g., developing → develop)

*ing → *e (e.g., saving → save)

*ying →*ie (e.g., die →dying)

s 结尾的动词单数第三人称;

*s → * (e.g., works → work)

*es → * (e.g., discuss → discusses)

*ies → *y (e.g., studies → study)

英文词形还原——Lemmatization

规则变化的词形还原

ly 结尾的副词

*ly → * (e.g., hardly → hard)

er/est 结尾的形容词比较级、最高级

er → (e.g., cold → colder)

*ier → *y (e.g., easier → easy)

s/ses/xes/ches/shes/oes/ies/ves 结尾的名词复数，**ies/ves** 结尾的名词还原时做相应变化**:**

bodies → body, shelves → shelf,

boxes → box, etc.

名词所有格**X's, Xs'**

英文词形还原——Lemmatization

不规则变化的动词、名词、形容词、副词的词形还原

choose, chose, chosen

bad, worse, worst

表示时间、百分数、货币、序数词的词形还原

1990s → 1990，标明时间名词；

82th →去掉th 后，记录该数字为序数词；

$200 → 去掉$，记录该数字为名词(200美圆)；

98.5% → 98.5%作为一个数词

英文词形还原——Lemmatization

合成词的词形还原

分数词：基数词和序数词

One-fourth。

合成名词：{名词、形容词、动词} + 名词

Human-computer, multiengine, large-scale 。

合成形容词：形容词 + 名词 + **ed**、形容词 + 现在分词、副词 + 现在分词、名词 + 过去分词、名词 + 形容词等

machine-readable, hand-coding, context-free

rule-based, speaker-independent 等。

英文词形还原——Lemmatization

合成词的词形还原

合成动词：{名词、形容词、副词} + 动词

Job-hunt

带连字符“-”的合成词

co-operate, 7-color, bi-directional, inter-lingua

Chinese-to-English,state-of-the-art, part-of-speech

OOV-words, text-to-speech, semi-automatically,*i*-th

英文词形还原——Lemmatization

词形还原的一般方法

查词典，如果词典中有该词，直接确定该词的原形；

查找不规则词形变化的词表，如果词典中有该词，直接确定该词的原形；

根据词形变化规则集，对单词进行还原处理，如果还原后在词典中找到该词，则得到该词的原形；

上述方法均失效，则作为未登录词处理。

英文词形还原的程度

词干层

如：impossibilities→ impossibility+ies

词根层

如：impossibilities → im+poss+ibil+it+ies

分析程度取决于自然语言处理系统的深度：

不解决未登录词，分析到词干层

解决未登录词，要分析到词根层。

code

词干还原

```
import nltk
from nltk.stem import
PorterStemmer
st = PorterStemmer()
wordlist =
['working','happiness']
```

[u'work',
u'happi']

```
import nltk
from nltk.stem import
LancasterStemmer
st = LancasterStemmer()
wordlist =
['working','happiness']
```

['work', u'happy

code

词型还原

```
import nltk
from nltk.stem import
WordNetLemmatizer
st = WordNetLemmatizer()
wordlist =
['working','happiness']
```

[u'working',
u'happines']

```
st.lemmatize('working',pos='v'
)
st.lemmatize('works')
```

u'work'
u'work'

中文的词法分析

中文的词法分析

分析型语言：

词之间没有边界标记。

词缺少形态变化。

词法分析

分词：将句子转换成词序列。

词性标注：将句子转换成词性序列。

中文的词法分析

重叠词、离合词、词缀处理

中文词的切分歧义——分词

中文未登录词识别

词性标注

中文词的重叠形式

形容词(AB)	ABAB式	AABB式	A里AB式
高兴	高兴高兴	高高兴兴	

动词(AB)	ABAB式	AABB式

动词（V）	VV式	V一V式	V了V式	V了一V式
听	听听	听一听	听了听	听了一听
想	想想	想一想	想了想	想了一想
玩	玩玩	玩一玩	玩了玩	玩了一玩
醒	醒醒	醒一醒		
试	试试	试一试	试了试	试了一试
笑	笑笑	笑一笑	笑了笑	笑了一笑
讲	讲讲	讲一讲	讲了讲	讲了一讲

中文其他词类的重叠形式

名词

哥哥，人人

山山水水，是是非非，方方面面，头头脑脑

数词

一一做了回答，两两结伴而来

量词

个个都是好样的，回回考满分

副词

常常，仅仅，的的确确

汉语重叠词处理

中文重叠词的重叠方式有很强的规律，处理起来并不困难。

中文的双字形容词的重叠现象主要有三种：

AABB、ABAB、A里AB。

这种词，只要还原成 原型AB，并查词典即可。

规则：

AABB→ AB

ABAB → AB

A里AB → AB

中文词缀

前缀

老鹰、老虎、老三、老王

超豪华、超标准、超高速

非党员

后缀

骨头、砖头、甜头、苦头、盼头、想头

桌子、椅子、孩子、票子、房子

文学家、指挥家、艺术家

科学性、可能性、学术性

碗儿、花儿、玩儿、份儿、片儿

采用规则处理

中文离合词

中文动词存在离合词现象

游泳：游了一会儿泳

理发：发理了没有

担心：担什么心

洗澡：洗了个热水澡

离合词的处理稍微复杂一些。一般的词法分析器都没有对离合词进行处理，只是把他们作为两个词对待。

离合词涉及到远距离的词约束关系。

中文分词

中文分词的重要性

分词是中文句法分析的基础

词的分析具有广泛的应用（词频统计，词典编纂，文章风格研究等）

文本处理主要以词为特征

“以词定字、以词定音”，用于文本校对、同音字识别、多音字辨识、简繁体转换

中文的切分歧义

交集型歧义（交叉型歧义）：

如果字串abc既可切分为ab/c，又可切分为a/bc。其中a，ab，c和bc是词

例子：我/对/他/有/意见/。总统/有意/见/他。

组合型歧义（覆盖型歧义）：

若ab为词，而a和b在句子中又可分别单独成词

例子：

马上：我/马上/就/来。他/从/马上/下来/。

将来：我/将来/要/上/大学。我/将/来/上海。

交集型歧义字段的链长

链长：交集型歧义字段中含有交集字段的个数，称为链长。

链长为1：和尚未

链长为2：结合成分

链长为3：为人民工作

链长为4：中国产品质量结合成分子时

链长为6：努力学习语法规则

链长为8：治理解放大道路面积水

真实语料中歧义字段的分布

刘开瑛，2000，《中文文本自动分词和标注》，商务印书馆，第65页。

（500万新闻语料的统计结果）

链长	1	2	3	4	5	6	7	8	总计
词次数	47402	28790	1217	608	29	19	2	1	78248
比例	50.58	47.02	1.56	0.78	0.04	0.02	0.00	0.00	100
字段数	12686	10131	743	324	22	5	2	1	23914
比例	53.05	42.36	3.11	1.35	0.09	0.02	0.01	0.01	100

中文分词标准

建立中文分词标准的必要性

- 汉语词定义不明确
  - 牛肉是词，鸡肉是不是？
  - 打倒是词，打死、打伤、饿死、涂黑是不是？
- 为操作的方便，必须确定统一的标准或规范
- 采用“分词单位”的说法

问题

- 取舍理由不够充分，人为色彩过重
- 过于复杂，难于把握

中文分词标准

相关的标准

《信息处理用汉语分词规范》

GB/T13715-92，中国标准出版社，1993

《资讯处理用中文分词规范》台湾中研院

《人民日报》语料库词语切分规范

……

中文分词标准

《人民日报》标注语料库词语切分规范（述补结构的切分）

未收入词典的双音节述补结构，若拆开各是一个词，通常作为两个切分单位。

走/v 到/v，撞/v 上/v，调/v 好/a，坐/v 稳/a

若拆开了，其中至少有一个是语素，通常就不切分，作为一个切分单位。

形成/v, 鼓动/v，说明/v，震动/v

双音节的述补结构中间插入“得”或“不”一般应予切分，

走/v 得/u 到/v，走/v 不/d 到/v，安/v 得/u 上/v，安/v 不/d 上/v

但是如果去掉“得”或“不”后，前后两个字不构成一个词的，则作为一个分词单位。

来得及/v，来不及/v，对得起/v，对不起/v，说得过去/l，说不过去/l

有的去掉“得”或“不”后虽然是一个合成词，但其中至少有一个是语素，拆开了却是难以理解的，仍作为一个切分单位

形得成/v，形不成/v

中文分词算法

- 规则分词
  - 最大匹配法
  - 最少分词法
  - 基于记忆的交叉歧义排除法
- 统计分词方法
  - 基于语言模型的切分算法
  - 基于HMM的切分
  - 其他统计模型

最大匹配法

- 最大匹配法（Maximum Matching, MM）
  - 有词典切分，机械切分
  - 正向最大匹配算法(Forward MM, FMM)
    - 自左往右
    - 每次取最长词
  - 逆向最大匹配算法(Backward MM, BMM)
    - 自右往左
    - 每次取最长词
  - 双向最大匹配算法（Bi-directional MM）
    - 依次采用正向和逆向最大匹配
    - 如果结果一致则输出
    - 如果结果不一致再用其他方法排歧

最大匹配法——正向最大匹配

基本思想：

输入字串S，假定分词词典中的最长词有i 个字符

将S的前i 个字作为匹配字段Q，查找字典。

若匹配成功，Q被作为一个词切分出来。

若匹配失败，将Q中的最后一个字去掉，对剩下的字串重新进行匹配处理。

如此进行下去，直到匹配成功，即切分出一个词或剩余字串的长度为零为止。这样就完成了一轮匹配

然后取下一个i 字字串进行匹配处理，直到文档被扫描完为止。

最大匹配法——正向最大匹配

S：“南京市长江大桥”，字典最大词长5

step1：1. 南京市长江大桥 ---- 匹配失败

2. 南京市长江大桥 ----匹配失败

3. 南京市长江大桥 ----匹配成功，切分出第一个词

南京市 长江大桥

step2：1. 南京市长江大桥 ---- 匹配失败

2. 南京市长江大桥 ---- 匹配失败

3. 南京市长江大桥 ---- 匹配成功，切分出第二个词

南京市 长江 大桥

。。。。。。

最大匹配法

逆向最大匹配法（RMM）——分词切分的方向与MM相反，从被处理文档的末端开始匹配扫描，每次取最末端的i 个字符作为匹配字段。相应地，它使用的分词词典是逆序词典，其中的每个词条都将按逆序方式存放。

双向最大匹配法——将正向和逆向最大匹配法得到的结果进行比较，然后按照最大匹配原则，选取词数切分最少的作为结果

中文中90.0% 左右的句子两种方法结果完全重合且正确，大概9 .0% 的句子两种切分方法得到的结果不一样，但其中必有一个是正确的(歧义检测成功) ，只有不到1.0 % 的句子，两种方法的切分虽重合却是错的，或者两个方法不同但两个都不对(歧义检测失败) 。这正是双向最大匹配法在实用中文信息处理系统中得以广泛使用的原因。

最大匹配法

优点

- 简单、快速
- 在某些应用场合已经足够

缺点

- 单向最大匹配会忽略交集型歧义和组合型歧义
  - 幼儿园/地/节目
  - 独立自主/和平/等/互利/的/原则
- 双向最大匹配会忽略链长为偶数的交集型歧义和组合型歧义
  - 原子/结合/成分/子时
  - 他/从/马上/下来

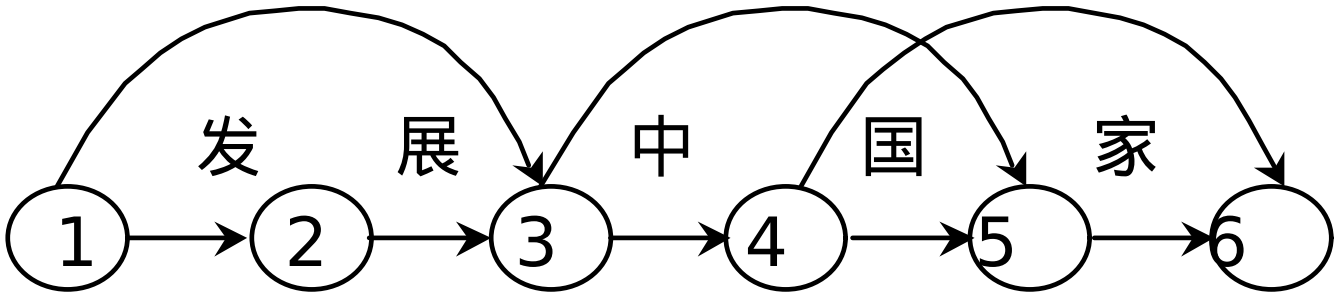
基于记忆的交叉歧义排除法

（孙茂松，1999）考察了一亿字的语料，发现交集型歧义字段的分布非常集中。其中在总共的22万多个交集型歧义字段中，高频的4,619个交集型歧义字段占所有歧义切分字段的59.20%。而这些高频歧义切分字段中，又有4,279个字段是伪歧义字段，也就是说，实际的语料中只可能出现一种切分结果。这样，仅仅通过基于记忆的方法，保存一种伪歧义切分字段表，就可以使交集型歧义切分的正确率达到53%，再加上那些有严重偏向性的真歧义字段，交集型歧义切分的正确率可以达到58.58%。

最少分词法

分词结果中含词数最少

等价于在有向图中搜索最短路径问题



最少分词法

基本思想：

在词图上选择一条词数最少的路径

算法：

动态规划算法

优点：好于单向的最大匹配方法

最大匹配：独立自主/和平/等/互利/的/原则

最短路径：独立自主/和/平等互利/的/原则

缺点：忽略了所有覆盖歧义，也无法解决大部分交叉歧义

结合成分子时

统计分词——基于N元语法的切分排歧

基于N元语法的切分排歧

$$\widehat{Seg} = \arg\max_{Seg} P(Seg \mid Text)$$

$$= \arg\max_{Seg} \frac{P(Text \mid Seg)P(Seg)}{P(Text)}$$

$$= \arg\max_{Seg} P(Text \mid Seg)P(Seg)$$

$$= \arg\max_{Seg} P(Seg)$$

$$P(Seg = P(w_1 / w_2 / \cdots / w_m) = P(w_1, w_2, \cdots, w_m) = \prod_{i=1}^{m} P(w_i \mid h)$$

统计分词step1——全切分

根据词典给出一个句子所有

例：对“中华人民共和国”

1、[中华人民共和国
2、[华人，　华]
3、[人民共和国，
4、[民]
5、[共和国，　共和，
6、[和]
7、[国]

1：[中华人民共和国]
2：[中华人民，　共和国]
3：[中华人民，　共和，　国]
4：[中华人民，　共，　和，　国]
5：[中华，　人民共和国]
6：[中华，　人民，　共和国]
7：[中华，　人民，　共和，　国]
8：[中华，　人民，　共，　和，　国]
9：[中华，　人，　民，　共和国]
10：[中华，　人，　民，　共和，　国]
11：[中华，　人，　民，　共，　和，　国]
12：[中，　华人，　民，　共和国]
13：[中，　华人，　民，　共和，　国]
14：[中，　华人，　民，　共，　和，　国]

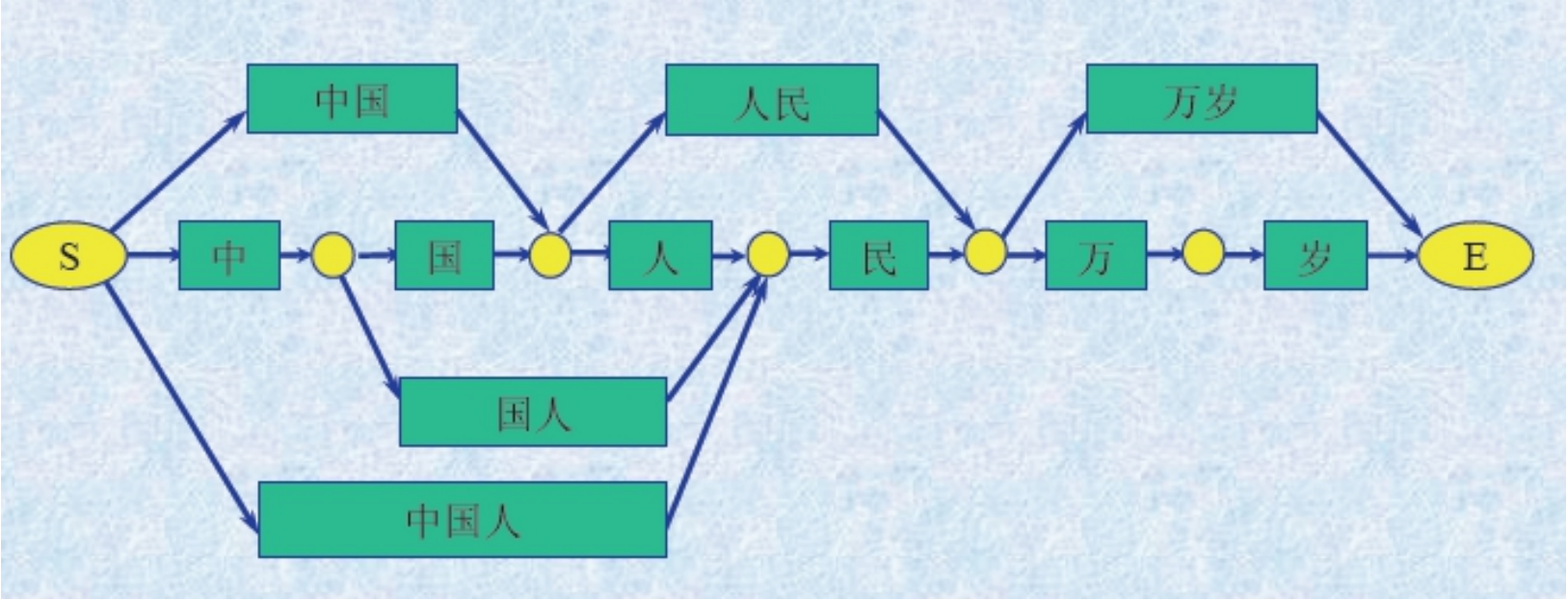
中文切分的数据结构－词图

根据这个数据结构，可以把词法分析中的几种操作转化为：

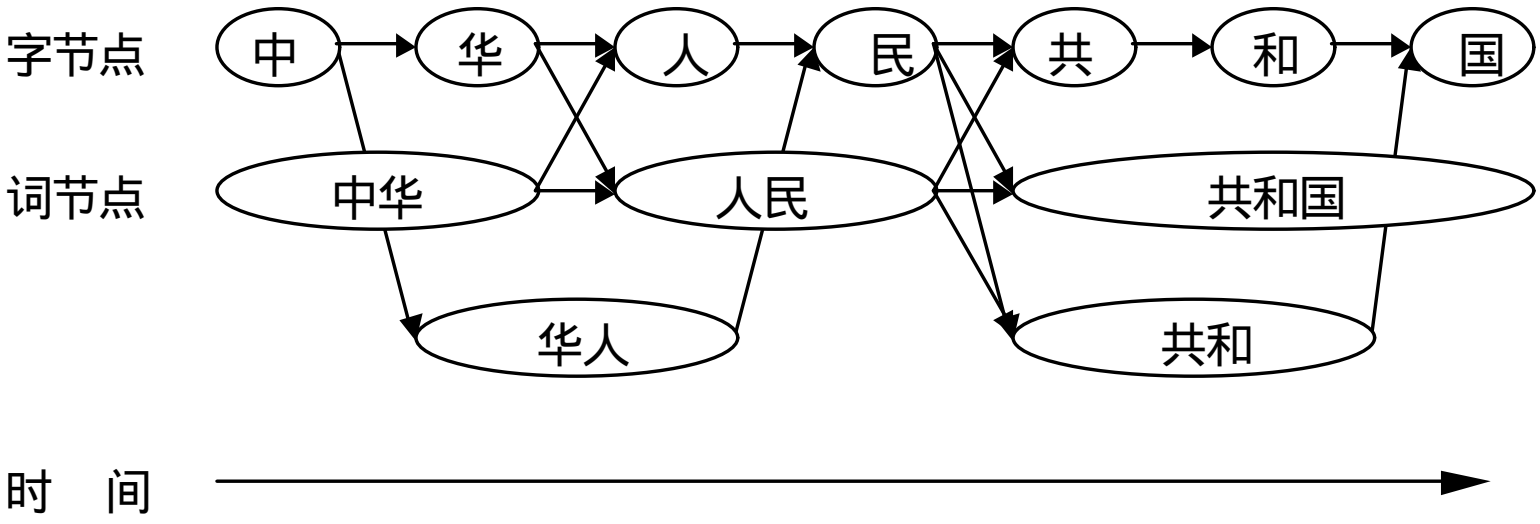
给词图上添加边（查词典，处理重叠词、离合词、前后缀和未登录词）；

寻找一条起点S到终点E的最优路径（切分排歧）；

给路径上的边加上标记（词性标注）；



字串“中华人民共和国”的切分词网格



基于统计的词网格分词

第一步是候选词网格构造：利用词典匹配，列举输入句子所有可能的切分词语，并以词网格形式保存

第二步计算词网格中的每一条路径的权值（n-gram模型），权值通过计算图中每一个节点（每一个词）的一元统计概率和节点之间的二元统计概率的相关信息

根据图搜索算法在图中找到一条权值最大的路径，作为最后的分词结果

动态规划算法：Viterbi算法

A*启发式搜索算法

基于N元语法的切分排歧

采用一元语法

即把切分路径上每一个词的词频相乘得到该切分路径的概率

把词频的负对数理解成“代价”，这种方法也可以理解为最少分词法的一种扩充

正确率可达到92%

简便易行，效果一般好于基于词表的方法

分词系统的评价

方法一：严格按照某种规范进行评价

算法简单

不够合理

方法二：允许一定程度的不同理解

“鸡肉”切成一个词或两个词都算正确

较为合理

算法与数据结构都较为复杂，或者引入人工测试数据的构造比较费时。

分词系统的评价

评价指标

精确率 P = 切分正确的词数/总的切分词数

召回率 R =切分正确的词数/正确的词的总数

F1值 F1 = 2*P*R/P+R

常见的词法分析工具

工具	功能	链接
Stanza	（python）分词、词性标注、句法分析、命名实体识别	https://stanfordnlp.github.io/stanza/index.html#about
CoreNLP	（Java）分词、词性标注、句法分析、命名实体识别、指代消解	https://stanfordnlp.github.io/CoreNLP/index.html
Paddle Lac	分词、词性标注、命名实体识别	https://aistudio.baidu.com/aistudio/projectdetail/305812
LTP	分词、词性标注、命名实体识别、依存句法分析、语义角色标注	http://www.ltp-cloud.com/
Jieba	分词、词性标注、关键词提取	https://github.com/fxsjy/jieba
SnowNLP	分词、词性标注、情感分析、文本分类、关键词提取，自动摘要相似度计算	https://pypi.python.org/pypi/snownlp/0.11.1
Hanlp	分词、词性标注、句法分析、关键词提取、自动摘要、命名实体识别、拼音转换等	http://hanlp.linrunsoft.com/

其他：中科院分词系统NLPIR，FoolNLTK，PKUSeg，Thulac

中文分词工具——jieba

特点：

社区活跃，使用简单，功能丰富，还包括关键词提取、词性标注等。

提供多种编程语言实现。Jieba 官方提供了Python 、c++ 、Go 、R 、iOS 等，在实际项目中，进行扩展十分容易。

https://github.com/fxsjy/jieba

原理：

统计规则结合

基于前缀词典的词图扫描，构建包含全部分词结果的有向无环图

基于统计语言模型找出概率最大的路径

基于HMM的未登录词识别

中文分词工具——jieba

```
#分词
>>>import jieba
>>> seg list = jieba.cut(sent , cut_all=True ) //全切模式
>>> seg_list = jieba.cut(sent) //精确模式
>>> seg_list = jieba . cut_ for_ search(sent) //搜索引擎模式
#添加自定义词典, file_name为自定义词典的路径，词典格式：每词一行；每行分三部分：词语，词频，词性（可省略），用空格隔开
>>> jieba.load_userdict(file_name)
# 词性标注
>>> import jieba.posseg as pseg
>>> words =pseg.cut("我爱北京天安门")
>>> for w in words:
        print w.word, w.flag
```

课堂练习

用jieba分词进行
下列句子的切分

研究生命科学

研究生命令本科生

我从马上下来

我马上下来

北京大学生喝进口红酒

在北京大学生活区喝进口红酒

从小学电脑

从小学毕业

美军中将竟公然说

新建地铁中将禁止商业摊点

这块地面积还真不小

地面积了厚厚的雪

原子结合成分子时

部分居民生活水平

治理解放大道路面积水

这样的人才能经受住考验

在这些企业中国有企业有十个

结婚的和尚未结婚的

中文未登录词识别

未登录词的类型

命名实体（Named Entity）

- 汉语人名：*李素丽 老张 李四 王二麻子*（较成熟）
- 汉语地名：*定福庄 白沟 三义庙 韩村 河马甸*（较成熟）
- 翻译人名：*乔治·布什 叶利钦 包法利夫人*（较成熟）
- 翻译地名：*阿尔卑斯山 新奥尔良 约克郡*（较成熟）
- 机构名：*方正公司 联想集团 国际卫生组织外贸部*（较困难）

数字、日期词、货币等（很成熟）

商标字号：*非常可乐 乐凯 波导 杉杉 同仁堂*（较困难）

专业术语：*万维网 主机板 贝叶斯算法*（很困难）

缩略语：*三个代表 五讲四美 打假扫黄*（很困难）

新词语：*新冠 人艰不拆 活久见 蓝瘦 香菇*（很困难）

未登录词识别的困难

未定义词没有明确边界

未定义词的构成单元（汉字）本身都可以独立成词

未登录词识别的一般方法

收集大量数据，建立不同的数据模型

常用的方法包括

规则方法：人工总结或归纳出一些判别规则（数字、日期、货币词等识别）

统计方法：建立统计模型，通过人工标注语料库进行参数训练

各种不同类型的未登录词识别方法思想大同小异，但实现时各有侧重

未登录词识别的依据

内部构成规律（用字规律）

外部环境（上下文）

重复出现规律

中国人名的内部构成规律

在汉语的未定义词中，中国人名是规律性最强，也是最容易识别的一类；

中国人名一般由以下部分组合而成：

姓：张、王、李、刘、诸葛、西门、范徐丽泰

名：李素丽，张华平，王杰、诸葛亮

前缀：老王，小李

后缀：王老，赵总

中国人名各组成部分用字比较有规律

中国人名的内部构成规律

台湾出版的《中国姓氏集》收集姓氏5544 个，其中，单姓3410 个，复姓1990 个，3字姓144 个。

中国目前仍使用的姓氏共737 个，其中，单姓729 个，复姓8 个。

根据我们收集的300 万个人名统计：姓氏：974个，其中，单姓952个，复姓23 个，300万人名中出现汉字4064个。

中国人名的内部构成规律

中国人名各组成部分的组合规律

- 姓 + 名
- 姓
- 名
- 前缀 + 姓
- 姓 + 后缀
- 姓 + 姓 + 名（海外已婚妇女）

中国人名的上下文构成规律

身份词：

前：工人、教师、影星、犯人

后：先生、同志

前后：女士、教授、经理、小姐、总理

地名或机构名：

前：静海县大丘庄禹作敏

的字结构

前：年过七旬的王贵芝

动作词

前：批评，逮捕，选举

后：说，表示，吃，结婚

中国人名识别的难点

- 一些高频姓名用字在非姓名中也是高频字
  - 姓氏：于，马，黄，张，向，常，高
  - 名字：周鹏和同学，周鹏和同学
- 人名内部相互成词，指姓与名、名与名之间本身就是一个已经被收录的词
  - [王国]维、[高峰]、[汪洋]、张[朝阳]
- 人名与其上下文组合成词
  - 这里[有关]天培的壮烈；
  - 费孝通向人大常委会提交书面报告
- 人名地名冲突
  - 河北省刘庄

中文姓名识别方法

中文姓名识别方法

姓名库匹配，以姓作为触发信息，寻找潜在的名字

计算潜在姓名的概率估值及相应姓氏的姓名阀值，根据姓名概率评价函数和修饰规则对潜在的姓名进行筛选。

中文姓名识别方法

设姓名 $Cname = Xm_1m_2$，其中 X 表示姓，m_1m_2 分别表示名字首字和名字尾字。

分别用下列公式计算姓氏和名字的使用频率：

$$F(X) = \frac{X\text{用作姓氏}}{\text{X出现的总次数}}$$

$$F(m_1) = \frac{m_1\text{作为名字首字出现的次 数}}{m_1\text{出现的总次数}}$$

$$F(m_2) = \frac{m_2\text{作为名字首字出现的次 数}}{m_2\text{出现的总次数}}$$

中文姓名识别方法

字串 $Cname$ 可能为姓名的概率估值：

$$P(Cname)=\begin{cases}F(X)\times F(m_1)\times F(m_2) & \text{复名情况}\\ F(X)\times F(m_2) & \text{单名情况}\end{cases}$$

姓氏 X 构成姓名的最小阀值：

$$T_{\min}(X)=\begin{cases}F(X)\times Min(F(m_1)\times F(m_2)) & \text{复名情况}\\ F(X)\times Min(F(m_2)) & \text{单名情况}\end{cases}$$

中文姓名识别方法

姓名的评价函数：

$$f = \ln P(Cname)$$

对于特定的姓氏 X 通过训练语料得到一阈值 β_X (threshold value)，当 f 大于 β_X 时，该识别的汉字串确定为中文姓名。

中国地名的识别

困难

地名数量大，缺乏明确、规范的定义。《中华人民共和国地名录》（1994）收集88026个，不包括相当一部分街道、胡同、村庄等小地方名称。

真实语料中地名出现情况复杂。如地名简称、地名用词与其它普通词冲突、地名是其它专用名词的一部分，地名长度不一等。

机构名的内部构成规律

机构名的内部构成规律

机构名一般都是定中结构。

机构名的后缀一般比较集中，识别相对容易。

机构名左边界识别非常困难。

机构名中含有大量的人名、地名、企业字号等专有名称。在这些专有名称中，地名所占的比例最大，其中未登录地名又占了相当一部分的比例。所以机构名识别应在人名、地名等其他专名识别之后进行，其他专名识别的正确率对机构名识别正确率有较大影响。

机构名的内部构成规律

中文机构名用词非常广泛。通过对人民日报1998年1月中的10817个机构名所含的19986个词进行统计，共计27种词，其中名词最多（9941个），地名其次（5023个），以下依次为简称（1169个）、专有名词（1125个）、动词（848个）以及机构名（714个）等

机构名长度极其不固定

机构名很不稳定。随着社会发展，新机构不断涌现，旧机构不断被淘汰、改组或更名。

中文机构名称的识别

中文机构名称的类型

地名，如：北京大学，武汉大学

人名，如：中山大学，哈佛大学

学科、专业、部门系统，如：公安部，教育委员会

研究、生产或经营等活动的对象，如：软件研究所，卫星制造厂

上述情况的综合，如：白求恩医科大学

机构名称识别方法

找到一机构称呼词

根据相应规则往前逐个检查名词作为修饰名词的合法性，直到发现非法词。

如果所接受的修饰词同机构称呼词构成一个合法的机构名称，则记录该机构名称。

统计模型

统计方法——识别问题转化成标注问题

在统计方法中，未登录词识别的一种最通常的做法就是将识别问题转化成标注问题

对于输入句子中的每个汉字，定义四个标记：

- 不属于未登录词O
- 未登录词首字B
- 未登录词尾字E
- 未登录词中间字I

汉字序列的标注问题可以采用隐马尔科夫模型（HMM）、最大熵（ME）、最大熵马尔科夫模型（MEMM）、条件随机场（CRF）等模型来解决

基于HMM的未登录词识别

以人名识别为例，输入文本：

这是周恩来、邓颖超生前居住的地方

标注为：

这是周恩来、邓颖超生前居住的地方

O O B I E O B I E O O O O O O O

两处标注为BIE的字串“周恩来”、“邓颖超”被识别为人名

训练语料库为已经标注人名的语料库

NER code for English text

```
# coding: utf-8
from nltk.tag import StanfordNERTagger
eng_tagger =
StanfordNERTagger('english.all.3class.distsim.crf.ser.gz')
print eng_tagger.tag('Rami Eid is studying at Stony Brook University in NY'.split())

[(u'Rami', u'PERSON'), (u'Eid', u'PERSON'), (u'is', u'O'), (u'studying', u'O'), (u'at', u'O'), (u'Stony', u'ORGANIZATION'), (u'Brook',
```

NER code for Chinese text

```
>>>pip install foolnltk
#依赖关系：python3.5 +, tensorflow> = 1.0.0
#coding:utf-8
>>> import fool
# 分词
>>> fool.load_userdict(path)
>>> print(fool.cut(text))
# 词性标注
>>> print(fool.pos_cut(text))
# 实体识别
>>> words, ners = fool.analysis(text)
>>>print(ners)
```

未登录词识别的评价

评价指标

精确率 P = 切分正确的词数/总的切分词数

召回率 R =切分正确的词数/正确的词的总数

F1值 F1 = 2*P*R/P+R

谢谢！

本课件参考了中科院计算所宗成庆研究员
的《自然语言理解》课件