如何运用stata进行文本分析

主讲人：张计宝

官方网站　stata-club.github.io

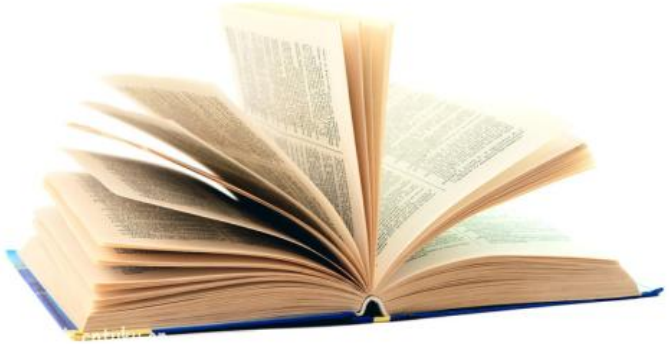
目录

一、分词原理

- 将一个汉字序列切分成一个一个单独的词

- This is a book. → ["This", "is", "a", "book", "."]

- 这是一本书。 → [("这", "pronoun"), ("是 ", "verb"),
("一", "numeral"), ("本", "classifier"),
("书", "noun"), ("。 ", "punctuation mark")]



为什么要进行中文分词？

- 词是最小的能够独立运用的语言单位

- 英文单词之间是以空格作为自然分界符

- 汉语是以字为基本的书写单位，词语之间没有明显的区分标记，因此，中文词语分析是中文信息处理的基础与关键。

- 武汉市长江大桥 → 武汉市 长江大桥
  武汉市 长江 大桥
  武汉 市长 江大桥

中文分词技术的分类

- 基于字典、词库匹配的分词方法
- 基于词频度统计的分词方法
- 基于知识理解的分词方法。

基于字典、词库匹配的分词方法

- 扫描方向的不同：
  - 正向匹配
  - 逆向匹配
- 长度优先匹配：
  - 最大（最长）匹配
  - 最小（最短）匹配

正向最大匹配法

- **正向最大匹配法 (Maximum Matching Method)**
- **简称为MM法**
- **基本思想**：假定分词词典中的最长词有i个汉字字符，则用被处理文档的当前字串中的前i个字作为匹配字段，查找字典。若字典中存在这样的一个i字词，则匹配成功，匹配字段被作为一个词切分出来。如果词典中找不到这样的一个i字词，则匹配失败，将匹配字段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。

- **分词字典**：爬虫俱乐部 全体成员 祝 Stata 大会 越来 越好

- **被处理文档**：爬虫俱乐部全体成员祝Stata大会越来越好

逆向最大匹配法

- 逆向最大匹配法(Reverse Maximum Matching Method)
- 简称为RMM法

- **基本原理：** 逆向最大匹配法从被处理文档的末端开始匹配扫描，每次取最末端的i字字串作为匹配字段，若匹配失败，则去掉匹配字段最前面的一个字，继续匹配。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。在实际处理时，先将文档进行倒排处理，生成逆序文档。然后，根据逆序词典，对逆序文档用正向最大匹配法处理即可。

- **分词字典：爬虫俱乐部 全体成员 祝 Stata 大会 越来 越好**

- **被处理文档：** 爬虫俱乐部全体成员祝Stata大会越来越好

例子

- 例如切分字段“硕士研究生产”
- 定义字典： 硕士研究生 产 硕士 研究 生产
- 正向最大匹配法的结果会是“硕士研究生/产”
- 逆向最大匹配法利用逆向扫描，可得到正确的分词结果“硕士/研究/生产”。

双向匹配法

- 将正向最大匹配法与逆向最大匹配法组合。
- 先根据标点对文档进行粗切分，把文档分解成若干个句子。
- 然后再对这些句子用正向最大匹配法和逆向最大匹配法进行扫描切分。如果两种分词方法得到的匹配结果相同，则认为分词正确。
- SunM.S和Benjamin K.T(1995)

基于词的频度统计的分词方法

- 基于词的频度统计的分词方法是一种全切分方法。
- **全切分**
- **基本思想**：要求获得输入序列的所有可接受的切分形式，而部分切分只取得一种或几种可接受的切分形式，由于部分切分忽略了可能的其他切分形式，所以建立在部分切分基础上的分词方法不管采取何种歧义纠正策略，都可能会遗漏正确的切分，造成分词错误或失败。而建立在全切分基础上的分词方法，由于全切分取得了所有可能的切分形式，因而从根本上避免了可能切分形式的遗漏，克服了部分切分方法的缺陷。
- **全切分缺点**：不具有歧义检测功能、导致庞大的无用数据充斥于存储数据库、,造成分词效率严重下降。

基于词的频度统计的分词方法

- **频度统计的分词方法的基本思想**:在上下文中，相邻的字同时出现的次数越多，就越可能构成一个词。因此字与字相邻出现的概率或频率能较好的反映词的可信度。

- **这是一种全切分方法**。它不依靠词典,而是将文章中任意两个字同时出现的频率进行统计,次数越高的就可能是一个词。它首先切分出与词表匹配的所有可能的词,运用统计语言模型和决策算法决定最优的切分结果。它的优点在于可以发现所有的切分歧义并且容易将新词提取出来。

基于知识理解的分词方法

- 该方法主要基于句法、语法分析，并结合语义分析，通过对上下文内容所提供信息的分析对词进行定界。

- 它通常包括三个部分：分词子系统、句法语义子系统、总控部分。在总控部分的协调下，分词子系统可以获得有关词、句子等的句法和语义信息来对分词歧义进行判断。

- 这类方法试图让机器具有人类的理解能力，需要使用大量的语言知识和信息。由于汉语语言知识的笼统、复杂性，难以将各种语言信息组织成机器可直接读取的形式。因此目前基于知识的分词系统还处在试验阶段。

目录

二、分词的实现

分词工具

- Stata中文分词系统
- Stata与python交互的中文分词系统
- Stata如何调用curl进行BosonNLP分词

Stata自定义分词程序

- moss命令的使用，主要运用匹配以及词频统计

```
clear
set obs 1
gen v = "this is a book"
moss v, match("is") //直接用is匹配，会匹配到两处。
moss v, match(" is ") prefix(a_) //is前后加空格会导致位置出错
moss v, match("(\bis\b)") regex unicode prefix(b_) //加上\b匹配单词边界，就不会出现错误
```

相对于中文有弊端，因为中文没有明显的分界符

主要使用函数 ustrwordcount() ustrword()

- ustrwordcount(s[,loc])

其中： s为字符串；locale 表示程序运行的不同语言环境

例如： “en”表示英文，“cn”表示中文，

每一个locale对象都代表了一个特定的地理、政治和文化地区。如果未指定locale，则使用默认语言环境，

例如： 这台电脑的操作系统是Microsoft Windows中文版，则系统默认语言环境设置为“cn”。

- 该函数返回的是字符串s中非空的unicode单词个数。

ustrword(s,n[,loc])

该函数返回的是字符串s中第n个位置的unicode单词。

其中： n为正数表示的是从s开头数第n个unicode单词，
n为负数表示的是从s结尾数第n个unicode单词。

例如： n为1表示的是返回s的第一个unicode单词，
n为-1表示的是返回s的最后一个unicode单词。
如果n大于cnt或小于-cnt，则该函数返回缺失值
其中cnt是s所包含的unicode单词数

jieba

jieba: 结巴(jieba)是国人出的一个精品插件，可以对一段中文进行分词，有三种分词模式，目前已有Python、JAVA、C++和Nodejs等版本。

jieba： https://github.com/fxsjy/jieba

- 全模式
- 精确模式
- 搜索引擎模式

Stata与python交互调用jieba进行分词

- **全模式**：将句子中所有的可以成词的词语都输出
- **例如**：输入文本“皮革厂”

  全模式下会将“皮革”、“皮革厂”都列出，速度非常快，但是不能解决歧义

  例如：“武汉市/长江大桥”和“武汉/市长/江大桥”。

- **精确模式**：试图将句子最精确地切开，会列出不重复的所有词，适合进行文本分析。

- **搜索引擎模式**在**精确模式**的基础上，对长词再次切分，适合用于搜索引擎分词。

- !pip install jieba

Stata如何调用curl进行BosonNLP分词

- BosonNLP API调用

http://docs.bosonnlp.com/tag.html

```
curl -X POST \
        -H "Content-Type: application/json" \
        -H "Accept: application/json" \
        -H "秘钥" \
        --data "\"\u6b66\u6c49\u5e02\u957f\u6c5f\u5927\u6865\""
        "http://api.bosonnlp.com/tag/analysis?space_mode=0&oov_level=3&t2s=0"
```

```
[{"word":["武汉市","长江","大桥"],"tag":["ns","ns","n"]}]
```

目录

词云图的实现

- 利用中文分词系统对文档分析
- 去除停用词
- 结合echart绘制词云图

词云图-剔除停用词

定义停用词表

常用停用词表：网上下载，包括标点符号、连词、介词、语气词等，项目停用词表：

根据项目需要添加

- ！＃＄％＆＇（）＊
- 上 上下 上前 上升 上去
- 不仅...而且 不仅仅 不但...而且

词云图再现


寿司
空调
充实
提交
调查表
打压
厚切牛舌
路过
装饰
吧台
适中
玉子烧
烟子
店面
习惯
份量
吃货
足够
延龄巷
小巷
好吃
蛋黄散
提前
十字路口
耳熟能详
性价比
整洁
墨鱼丸
食欲
算是
可爱
坚持不懈
群光
好事
拉面
Joe
牛肉
感觉
鳗鱼
发现
网红
虾滑
万岁
市场
服务
人气
天白
位置
日式
整体
商铺
新鲜
甜品
CNN
站
帮忙
地方
细腻
自然
清爽
海路
环境
不好
机会
名气
无限
刘某
low
兑换
喜欢
禾绿
肉馅
老板
个券
朋友
味道
食堂
裤脚
忘记
齐全
可惜
干净
帅哥
感谢
酥脆
九州
风格
健康
生意
充电
食材
店家
蔬菜
显眼
速度
菜式
特色
家居
完美
国际
服务员
价格
情况
倩倩
厨师
团购
甜点
中心
简单
武汉
饺子
皮子
温泉
骨汤
火焰
靠近
推荐
菜品
档次
酒窝
预约
太平南路
类似
态度
饱满
美味
油腻
小伙伴
店糖丘

文本案例分析

寻找地名——文本分析

冷面兄，你来给想个法子。”最后那句话，却是向冷面先生冷谦说的。冷谦嗯了一声，并不答话，他知彭和尚定要细问端详，自己大可省些精神。果然彭和尚一连串问话连珠价迸将出来，周颠说话偏又颠三倒四，待得说完经过，说不得和铁冠道人也已运气完毕。彭和尚与冷谦运起内力，分别为韦一笑、周颠驱除寒毒。

待得韦周二人元气略复。彭和尚道：“我从东北方来，得悉少林派掌门空闻亲率师弟空智、空性，以及诸代弟子百余人，正赶来光明顶参与围攻我教。”

说得不错，义父在冰火岛上一住二十年，未必肯以垂暮之年，重归中土，说道：“大海中风波无情，你何必去冒这个险？”

赵敏道：“你冒得险，我为甚么便不成？”张无忌踌躇道：“你爹爹肯放你去吗？”赵敏道：“爹爹叫我统率江湖群豪，这几年来我往东到西，爹爹从来就没管我。”

张无忌听到“爹爹叫我统率江湖群豪”这句话，心中一动：“我到冰火岛去迎接义父，不知何年何月方归。倘若那是她的调虎离山之计，乘我不在，便大举对付本教，倒是不可不防，若是和她

部分程序

程序如下：

```
clear
cap mkdir D:\金庸小说倚天屠龙记地名/添加字典进行分词
cd D:\金庸小说倚天屠龙记地名/添加字典进行分词
forvalues z = 1/40 {
    use "d:/金庸小说倚天屠龙记地名/第`z'章.dta",clear
    local a = _var1[1]  //局部宏`a'存储了某一章的内容。
    di "`a' "
    clear all
    tempname handle //使用临时句柄和file命令写一个python程序。
    file open `handle' using 分词.py, replace text write
    file write `handle' `"# -*- coding: utf-8-*-"' _n
    file write `handle' `"import jieba.posseg"' _n
    file write `handle' `"jieba.load_userdict("D:/金庸小说地名/金庸小说地名_分词
字典.txt")"' _n  //添加用户自定义的分词字典。
    file write `handle' `"string = "`a'""' _n  //局部宏`a'在这里调用，表示对`a'
里边的内容进行分词。
    file write `handle' `"seg_list = jieba.posseg.cut(string)"' _n
    file write `handle' `"for word in seg_list:"' _n
    file write `handle' `"      print(word.word, word.flag)"'    //打印出分词结
果，分词结果中包含关键词和词性。
    file close `handle'
    ! python 分词.py > 第`z'章_分词结果.txt  //在stata中调用python，运行py格式的
文件。把分词结果保存到txt文件中。
}
```

地点词云图


少林寺
光明顶
峨嵋
波斯
蒙古
武当山
龙门镖局
西域
中原
冰火岛
灵蛇岛
万安寺
蝴蝶谷
华山
昆仑山
罗汉堂
青海
临安
少室山
王盘山
江南
洛阳
襄阳
汉水
卢龙

高亮输出文本数据

- 举个例子 将文档中出现**张无忌、赵敏、倚天剑、屠龙刀**等词**高亮输出**

张无忌听得群丐去远，庙中再无半点声响，于是从鼓中跃了出来。赵敏跟着跃出，理一理身上衣衫，似喜似嗔地横了他一眼。张无忌怒道：“哼，亏你还有脸来见我？”赵敏俏脸一沉，道：“怎么啦？我甚么地方得罪张大教主啦？”张无忌脸上如罩严霜，喝道：“你要盗那倚天剑和屠龙刀，我不怪你！你将我抛在荒岛之上，我也不怪你！可是殷姑娘已然身受重伤，你何以还要再下毒手！似你这等狠毒的女子，当真天下少见。”说到此处，悲愤难抑，跨上一步，左右开弓，便是四记耳光。赵敏在他掌力笼罩之下，如何闪避得了？啪啪啪啪四声响过，两边脸颊登时红肿。赵敏又痛又怒，珠泪滚滚而下，哽咽道：“你说我盗了倚天剑和屠龙刀，是谁见来？谁说我对殷姑娘下了毒手，你叫她来跟我对质。”

高亮输出程序

```
import delimited using E:\爬虫俱乐部\文本分析\dict.txt, clear ///
    encoding("gb18030") varname(nonames)
levelsof v1, local(keyword) clean
disp "`keyword'"
restore
gen count=.
forval position=1/`N'{
    local word1=word[`position']
    if ustrregexm("`keyword'","\b`word1'\b"){
        replace count=1 in `position'
    }
}
replace count=0 if count==.
save E:\爬虫俱乐部\文本分析/分词结果,replace
local N=_N
forval i=1/`N'{
    use E:\爬虫俱乐部\文本分析/分词结果,clear
    local a=word[`i']
    local b=count[`i']
    if "`b'" == "1" {
```

高亮输出结果

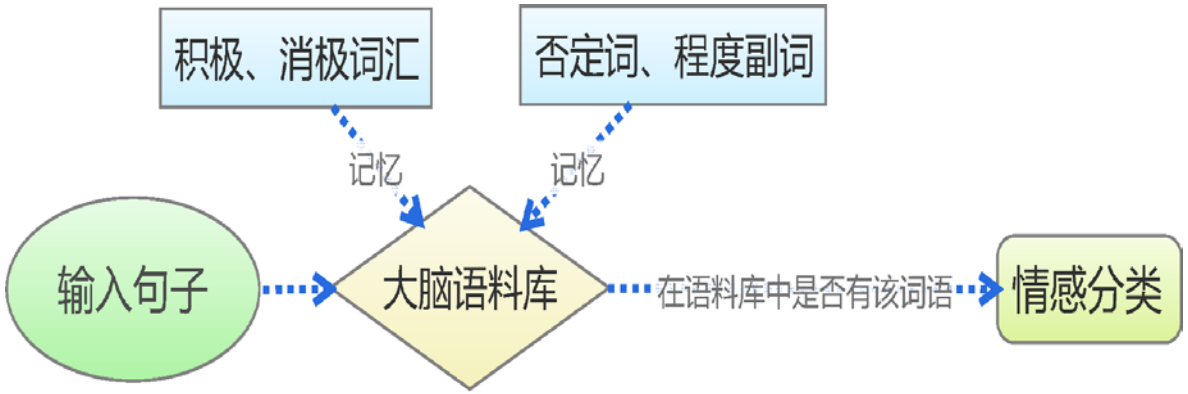
张无忌听得群丐去远，庙中再无半点声响，于是从鼓中跃了出来。赵敏跟着跃出，理一理身上衣衫，似喜似嗔地横了他一眼。张无忌怒道："哼，亏你还有脸来见我？"赵敏俏脸一沉，道："怎么啦？我甚么地方得罪张大教主啦？"张无忌脸上如罩严霜，喝道："你要盗那倚天剑和屠龙刀，我不怪你！你将我抛在荒岛之上，我也不怪你！可是殷姑娘已然身受重伤，你何以还要再下毒手！似你这等狠毒的女子，当真天下少见。"说到此处，悲愤难抑，跨上一步，左右开弓，便是四记耳光。赵敏在他掌力笼罩之下，如何闪避得了？啪啪啪啪四声响过，两边脸颊登时红肿。赵敏又痛又怒，珠泪滚滚而下，哽咽道："你说我盗了倚天剑和屠龙刀，是谁见来？谁说我对殷姑娘下了毒手，你叫她来跟我对质。"↵

目录

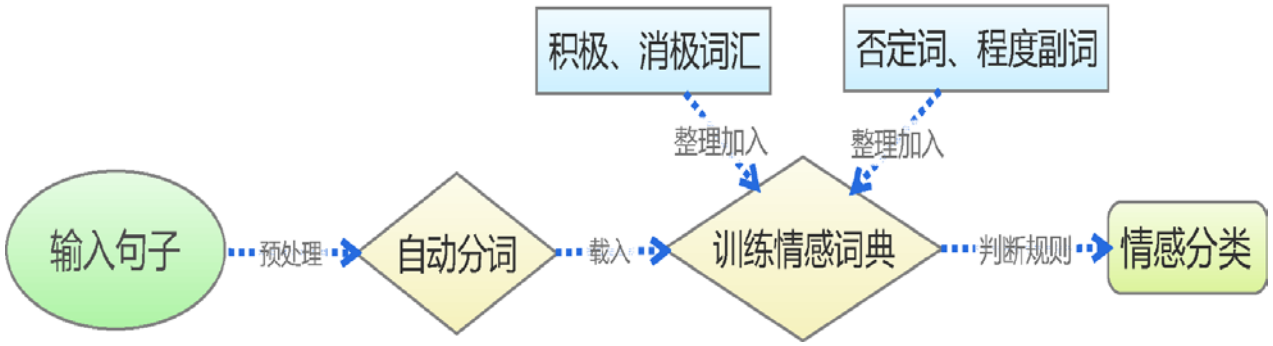
四、情感分析及其实现

- **情感分析**又称倾向性分析、意见挖掘。目的是为了找出说话者/作者在某些话题上或者针对一个文本两极的观点的态度。

- **大脑处理流程**



- **机器处理流程**



基于语调词典

- 语调分析
- 目前语调分析，最主要是关于中国上市公司业绩说明会这一信息披露平台及基于其数据的研究，仅谢德仁和林乐（2015，2016）两位学者对此进行了研究，他们在中文自动分词的基础上，利用词汇匹配技术法（也称为“词袋”方法），借鉴Henry（2008）、Henry and Leone（2009）等的做法，构建了管理层语调的指标，并对其是否具有信息含量作了实证检验。
- 结果发现，业绩说明会上的正面（/负面）的管理层语调预示着良好（/不良）的未来业绩并且产生了积极（/消极）市场反应

- 语调计算采用Price等（2012）、谢德仁等（2016）的简单比例加总权重的方法作为主要衡量方法，本文构建语调指标如下：

$$tone_{i,t} = \frac{postone_{i,t} - negtone_{i,t}}{postone_{i,t} + negtone_{i,t}}$$

谢 谢