# 用结构化数据的方式来管理文本

张丹

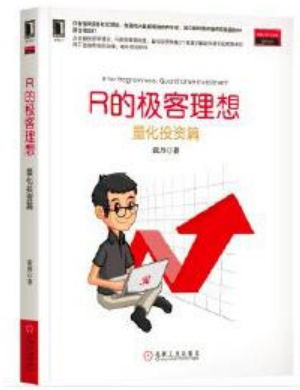
# 个人介绍

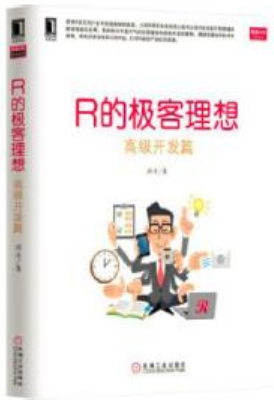
**张丹**， R语言实践者，北京青萌数海科技有限公司CTO，微软MVP。

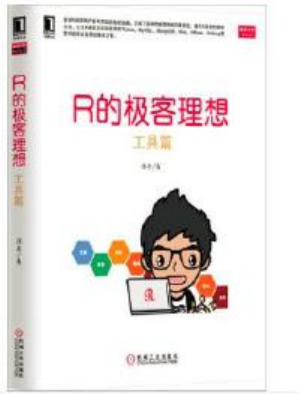
10年以上互联网应用架构经验，在R、Java、NodeJS、大数据、数据挖掘等方面有深厚的积累。

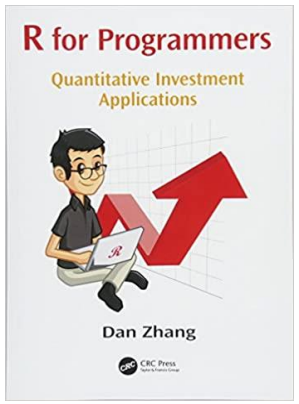
精通量化投资交易策略，熟悉中国金融二级市场、交易规则和投研体系。 熟悉数据学科方法论，在外汇、海关、区块链等领域均有落地的应用。

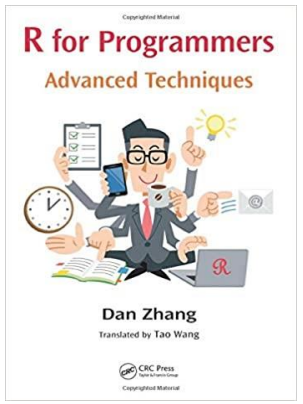
著有《R的极客理想：量化投资篇》、《R的极客理想：工具篇》、《R的极客理想：高级开发篇》，英文版图书被CRC出版集团引进，在美国发行。个人博客：http://fens.me 。

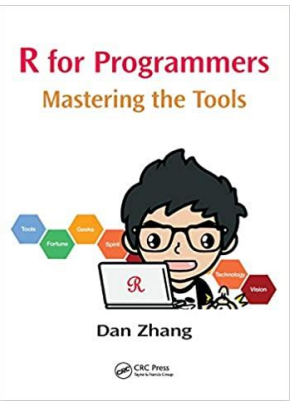


# 业务背景

在互联网的今天，我们每天都会生产和消费大量的文本信息，如报告、文档、新闻、聊天、图书、小说、语音转化的文字等。海量的文本信息，不仅提供扩宽的研究对象和研究领域，也为商业使用带来了巨大的机会。

量化文本分析（Quantitative Analysis of Textual Data），一种新的方式，用结构化数据的方式来管理文本。quanteda包，提出以语料库的形式管理文本，语料库被定义为文本的集合，其中包括特定每个文本的文档级变量，和整个集合的元数据。用户可以轻松地按单词、段落、句子甚至用户提供的分隔符分割文本和标签，按文档级变量将它们分组为更大的文档，形成基于逻辑条件的变量组合。

# jiebaR 中文分词

# jiebaR是什么?

结巴分词(jiebaR)，是一款高效的R语言中文分词包，底层使用的是C++，通过Rcpp进行调用很高效。结巴分词基于MIT协议，就是免费和开源的，感谢国人作者的给力支持，让R的可以方便的处理中文文本。

官方Github的地址：https://github.com/qinwf/jiebaR

# 5分钟上手

```
~ R
> install.packages("jiebaR")
> library("jiebaR")
```

对字符串进行分词

```
> wk = worker()

> wk["我是《R的极客理想》图书作者"]
[1] "我是" "R"    "的"   "极客" "理想" "图书" "作者"

> wk["我是R语言的深度用户"]
[1] "我"   "是"   "R"    "语言" "的"   "深度" "用户"
```

对文件进行分词

```
> wk['./idea.txt']
[1] "./idea.segment.2016-07-20_23_25_34.txt"
```

```
~ notepad idea.segment.2016-07-20_23_25_34.txt


R 的 极客 理想 系列 文章 涵盖 了 R 的 思想 使用 工具 创新 等 的
```

# 分词引擎

在调用worker()函数时，我们实际是在加载jiebaR库的分词引擎。jiebaR库提供了7种分词引擎。

- 混合模型(MixSegment):是四个分词引擎里面分词效果较好的类，结它合使用最大概率法和隐式马尔科夫模型。
- 最大概率法(MPSegment) :负责根据Trie树构建有向无环图和进行动态规划算法，是分词算法的核心。
- 隐式马尔科夫模型(HMMSegment):是根据基于人民日报等语料库构建的HMM模型来进行分词，主要算法思路是根据(B,E,M,S)四个状态来代表每个字的隐藏状态。 HMM模型由dict/hmm_model.utf8提供。分词算法即viterbi算法。
- 索引模型(QuerySegment):先使用混合模型进行切词，再对于切出来的较长的词，枚举句子中所有可能成词的情况，找出词库里存在。
- 标记模型(tag)
- Simhash模型(simhash)
- 关键词模型(keywods)

# 核心函数

```
worker(type = "mix", dict = DICTPATH, hmm = HMMPATH, user = USERPATH,
  idf = IDFPATH, stop_word = STOPPATH, write = T, qmax = 20, topn = 5,
  encoding = "UTF-8", detect = T, symbol = F, lines = 1e+05,
  output = NULL, bylines = F, user_weight = "max")
```

- type, 引擎类型
- dict, 系统词典
- hmm, HMM模型路径
- user, 用户词典
- idf, IDF词典
- stop_word, 关键词用停止词库
- write, 是否将文件分词结果写入文件，默认FALSE
- qmax, 最大成词的字符数，默认20个字符
- topn, 关键词数,默认5个
- encoding, 输入文件的编码，默认UTF-8
- detect, 是否编码检查，默认TRUE
- symbol, 是否保留符号，默认FALSE
- lines, 每次读取文件的最大行数，用于控制读取文件的长度。大文件则会分次读取。
- output, 输出路径
- bylines, 按行输出
- user_weight, 用户权重

```
> wk = worker()
> wk
Worker Type:  Jieba Segment

Default Method  :  mix      # 混合模型
Detect Encoding :  TRUE     # 检查编码
Default Encoding:  UTF-8    # UTF-8
Keep Symbols    :  FALSE    # 不保留符号
Output Path     :           # 输出文件目录
Write File      :  TRUE     # 写文件
By Lines        :  FALSE    # 不行输出
Max Word Length :  20       # 最大单单词长度
Max Read Lines  :  1e+05    # 最大读入文件行数

Fixed Model Components:

$dict                       # 系统词典
[1] "D:/tool/R-3.2.3/library/jiebaRD/dict/jieba.dict.utf8"

$user                       # 用户词典
[1] "D:/tool/R-3.2.3/library/jiebaRD/dict/user.dict.utf8"

$hmm                        # 隐式马尔科夫模型模型
[1] "D:/tool/R-3.2.3/library/jiebaRD/dict/hmm_model.utf8"

$stop_word                  # 停止词，无
NULL

$user_weight                # 用户词典权重
[1] "max"

$timestamp                  # 时间戳
[1] 1469027302
```

# 配置词典

```
# 查看默认的词库位置
> show_dictpath()
[1] "D:/tool/R-3.2.3/library/jiebaRD/dict"

# 查看目录
> dir(show_dictpath())
[1] "D:/tool/R-3.2.3/library/jiebaRD/dict"
 [1] "backup.rda"       "hmm_model.utf8"  "hmm_model.zip"
 [4] "idf.utf8"         "idf.zip"         "jieba.dict.utf8"
 [7] "jieba.dict.zip"   "model.rda"       "README.md"
[10] "stop_words.utf8" "user.dict.utf8"
```

看到词典目录中，包括了多个文件。

- jieba.dict.utf8, 系统词典文件，最大概率法，utf8编码的
- hmm_model.utf8, 系统词典文件，隐式马尔科夫模型，utf8编码的
- user.dict.utf8, 用户词典文件，utf8编码的
- stop_words.utf8，停止词文件，utf8编码的
- idf.utf8，IDF语料库，utf8编码的
- jieba.dict.zip，jieba.dict.utf8的压缩包
- hmm_model.zip，hmm_model.utf8的压缩包
- idf.zip，idf.utf8的压缩包
- backup.rda，无注释
- model.rda，无注释
- README.md，说明文件

# 用户词典

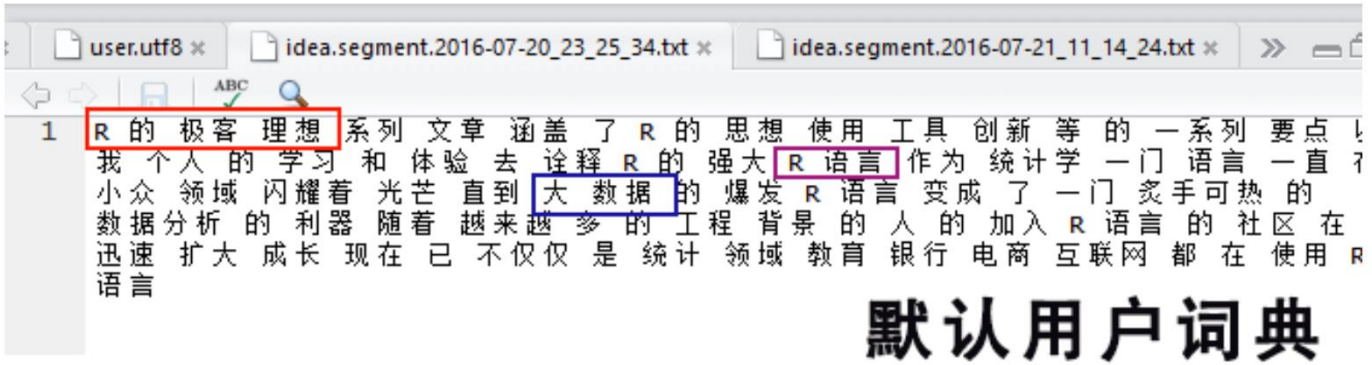
## 新建用户词典

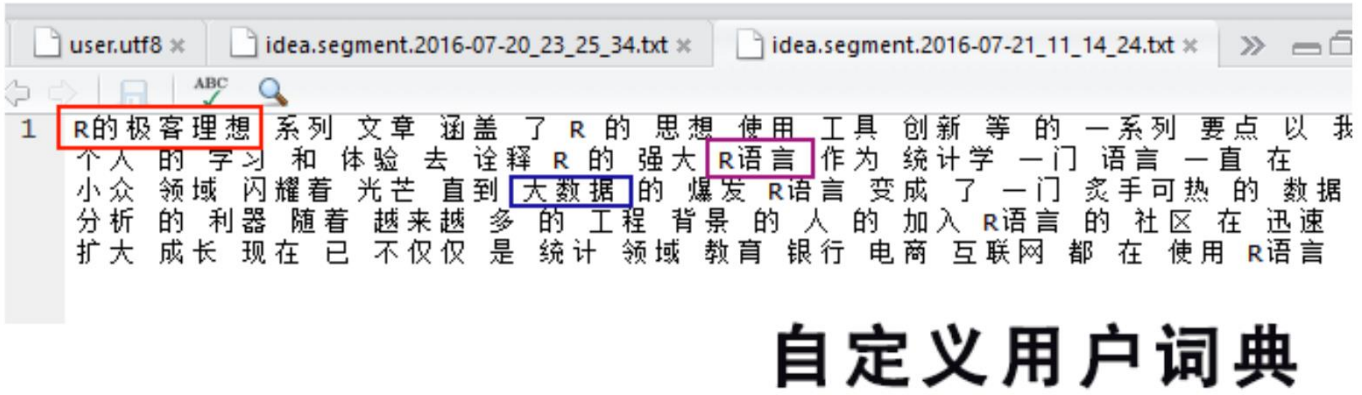
```
` notepad user.utf8

R语言
R的极客理想
大数据
数据
```

```
> wk = worker(user='user.utf8')
> wk['./idea.txt']
[1] "./idea.segment.2016-07-21_11_14_24.txt"
```

## 对比用户词典和默认词典



# 停止词过滤

停止词就是分词过程中，我们不需要作为结果的词，像英文的语句中有很多的a,the,or,and等，中文语言中也有很多，比如 的，地，得，我，你，他。这些词因为使用频率过高，会大量出现在一段文本中，对于分词后的结果，在统计词频的时候会增加很多的噪音，所以我们通常都会将这些词进行过滤。

```
~ notepad stop_word.txt


我
我是
```

## stop_word 参数

```
> wk = worker(stop_word='stop_word.txt')
> segment<-wk["我是《R的极客理想》图书作者"]
> segment
[1] "R"    "的"    "极客" "理想" "图书" "作者"
```

## filter_segment函数

```
> filter<-c("作者")
> filter_segment(segment,filter)
[1] "R"    "的"    "极客" "理想" "图书"
```

# 关键词提取

关键词提取是文本处理非常重要的一个环节，一个经典算法是TF-IDF算法。其中，TF（Term Frequency）代表词频，IDF（Inverse Document Frequency）表示逆文档频率。如果某个词在文章中多次出现，而且不是停止词，那么它很可能就反应了这段文章的特性，这就是我们要找的关键词。再通过IDF来算出每个词的权重，不常见的词出现的频率越高，则权重越大。

```
TF-IDF = TF(词频) * 逆文档频率(IDF)
```

## 自定义文本的关键字提取

```
> wk = worker()
> segment<-wk["R的极客理想系列文章，
# 计算词频
> freq(segment)
    char freq
1   创新    1
2     了    1
3   文章    1
4   强大    1
5      R    3
6   个人    1
7     的    5
8   诠释    1
9     和    1
10 一系列    1
11  使用    1
12    以    1
13    等    1
```

```
# 取TF-IDF的前5的关键词
> keys = worker("keywords",topn=5)

# 计算关键词
> vector_keywords(segment,keys)
11.7392 8.97342 8.23425  8.2137 7.43298
 "极客"  "诠释"  "要点"  "涵盖"  "体验"
```

## 词典语料

```
> scan(file="D:/tool/R-3.2.3/library/jiebaRD/dict/idf.utf8",
+      what=character(),nlines=50,sep='\n',
+      encoding='utf-8',fileEncoding='utf-8')
Read 50 items
 [1] "劳动防护 13.900677652"      "生化学 13.900677652"
 [3] "奥萨贝尔 13.900677652"      "考察队员 13.900677652"
 [5] "岗上 11.5027823792"         "倒车档 12.2912397395"
 [7] "编译 9.21854642485"         "蝶泳 11.1926274509"
 [9] "外委 11.8212361103"         "故作高深 11.9547675029"
[11] "尉遂成 13.2075304714"       "心源性 11.1926274509"
[13] "现役军人 10.642581114"      "杜勃留 13.2075304714"
[15] "包天笑 13.900677652"        "贾政陪 13.2075304714"
[17] "托尔湾 13.900677652"        "多瓦 12.5143832909"
[19] "多瓣 13.900677652"          "巴斯特尔 11.598092559"
[21] "刘皇帝 12.8020653633"       "亚历山德罗夫 13.2075304714"
[23] "社会公众 8.90346537821"     "五百份 12.8020653633"
[25] "两点阈 12.5143832909"       "多瓶 13.900677652"
[27] "冰天 12.2912397395"         "库布齐 11.598092559"
[29] "龙川县 12.8020653633"       "银燕 11.9547675029"
[31] "历史风貌 11.8212361103"     "信仰主义 13.2075304714"
```

# quanteda 量化文本分析

# Quanteda是什么?

Quanteda是一个用于管理和分析文本数据的 R包。

https://quanteda.io/

Quanteda从底层开始重新设计了文本处理过程，在语法与性能上得到了巨大提升。

- 内部使用stringi作为字符处理工具
- 内部基于data.table与Matrix包
- 统一的语法结构

Quanteda Initiative 总部位于伦敦，是一家英国非营利组织，致力于推广开源文本分析软件。主要产品R包 quanteda, readtext, spacyr, stopwords 。

https://quanteda.org/

# 核心函数

所述quanteda软件包由几个核心数据类型，通过调用具有相同名称的构造器创建的。

核心对象类型及其构造函数：

- corpus： 建立语料库对象
- tokens : 构造一个分词对象
- dfm : 创建文档特征矩阵
- fcm : 创建特征共现矩阵
- kwic : 关键字查询
- dictionary : 创建字典

# 文本处理过程

quanteda 文本重新定义了文本处理的过程，自己负责底层文本数据结构，结合应用层不同的功能包进行扩展。

配合使用的其他包：

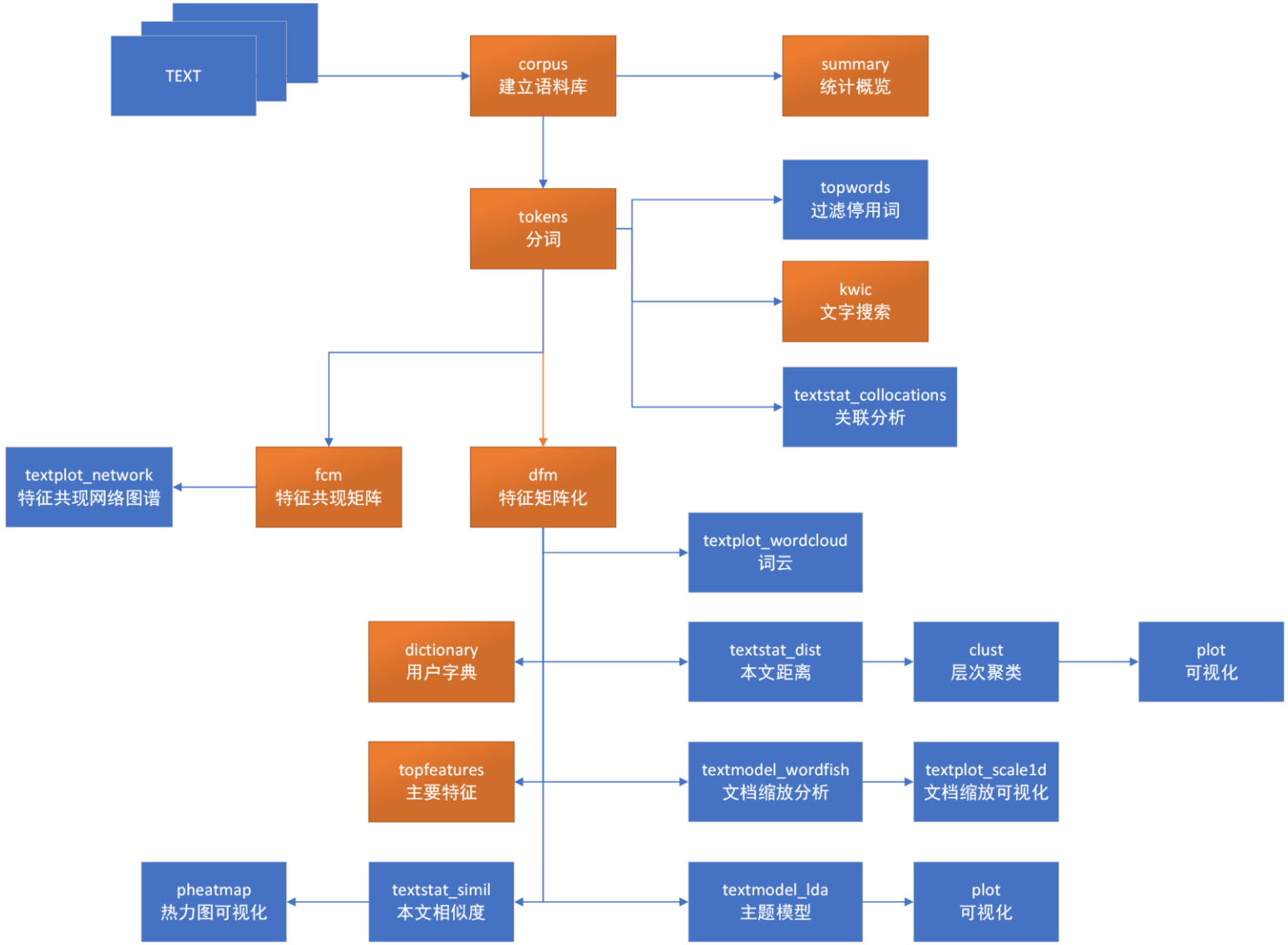
readtext, stopwords

quanteda.textstats

quanteda.textmodels

quanteda.textplots

ggplot2,magrittr,stringr,plyr,dplyr,reshape2,seededlda



# 文字处理包的对比

本文将quanteda包与用于定量文本分析的替代 R 包对比：tm、tidytext、corpus和koRpus

| Function | quanteda | tm | tidytext | corpus | koRpus | NLTK |
|---|---|---|---|---|---|---|
| Create corpus | corpus() | Corpus() | | corpus_frame() | read.corp.custom() | PlaintextCorpusReader() |
| Bind/subset corpora | corpus_subset() | tm_combine(); tm_filter() | | | | |
| Reshape corpus into smaller units | corpus_reshape(); corpus_segment() | | | text_split() | | |
| Take random sample of corpus texts | corpus_sample() | | | | | |
| Keywords-in-context | kwic() | | | text_locate() | | common_contexts() |
| Tokenize texts | tokens() | tokenizer() | unnest_tokens() | text_tokens() | tokenize() | nltk.word_tokenize |
| Stem features | tokens_wordstem() | stemDocument() | | stem_snowball() | treetag() | stem() |
| Define multi-word features | phrase() | | | | | MWETokenizer |
| Create document-feature matrix | dfm() | TermDocumentMatrix() | cast_dfm() | term_matrix() | | |
| Create a feature co-occurrence matrix | fcm() | | | | | |
| Weight a dfm | dfm_weight() | weightTf(); weightTfIdf() | bind_tf_idf() | | | |
| Create a custom dictionary | dictionary() | | dictionary always a data.frame object | | | SentimentAnalyzer |
| Included dictionaries | Lexicoder Sentiment Dictionary | | AFINN, Bing, NRC | AFINN Sentiment dictionary, WordNet-Affect Lexicon | | |

# 以职位数据为例子

从招聘网站上，只搜索 数据分析师 职位，进行数据获取。

字段：职位，薪资范围，薪数，职位描述。

# 建立语料库

```
# 加载数据集
docs<-readtext("./job/*.txt",
               docvarsfrom = "filenames",
               encoding = "UTF-8") # 公司，待遇，地点，职位，时间

###############
# 建立语料
###########
doc.corpus
doc.corpus
summary(dc

# 增加文档
docvars(dc
docvars(dc
docvars(dc
docvars(dc
salary<-st
docvars(dc
docvars(dc
```

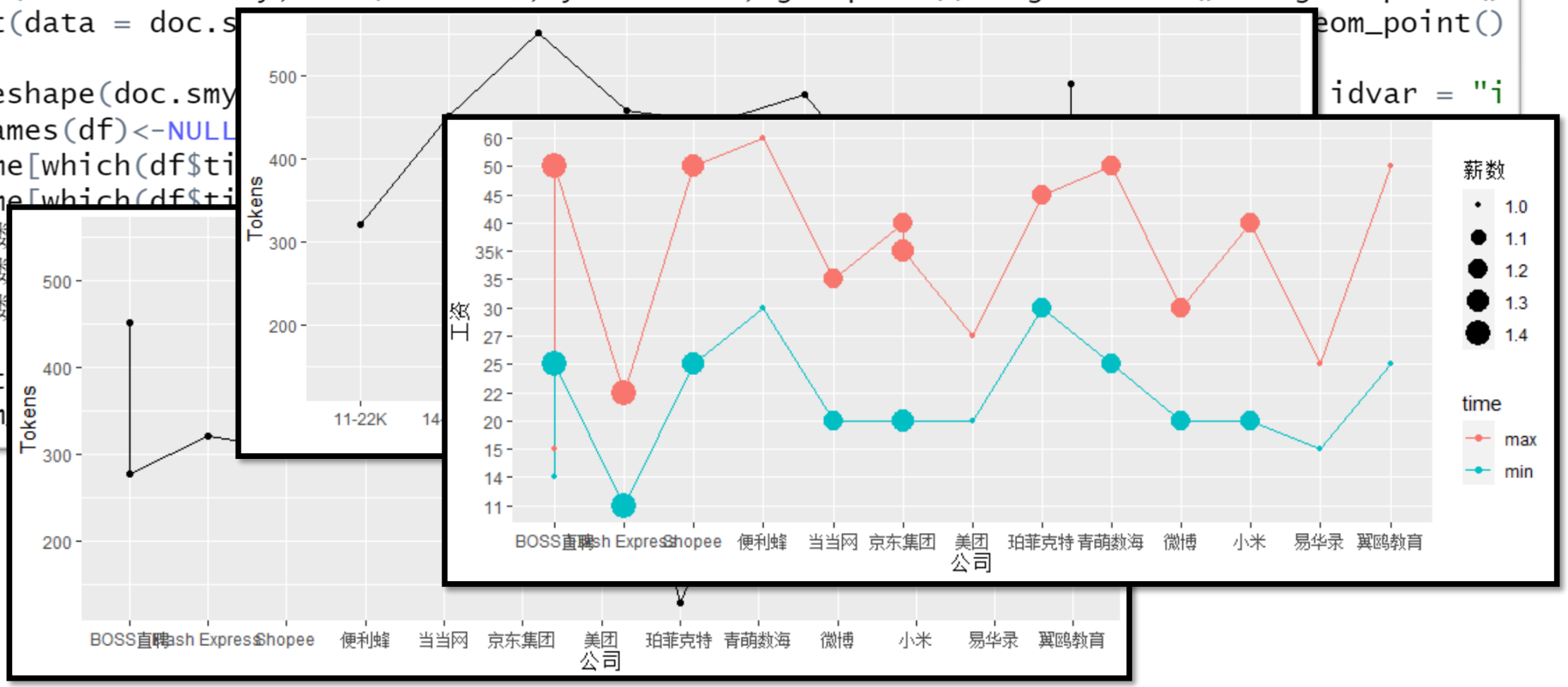
```
> doc.smy<-summary(doc.corpus);doc.smy
Corpus consisting of 15 documents, showing 15 documents:

   Text Types Tokens Sentences docvar1          公司        职位    工资 薪数 工资min 工资max
  1.txt   187    345        16       1      青萌数海 数据分析师 25-50K   14      25      50
  2.txt   202    346        11       2       当当网 数据分析师 20-35K   14      20      35
  3.txt   267    552        15       3       易华录 数据分析师 15-25K           15      25
  4.txt   233    452         3       4    BOSS直聘 数据分析师 14-15K           14      15
  5.txt   142    296         1       5      京东集团 数据分析师 20-40K   14      20      40
  6.txt   233    432         8       6        小米 数据分析师 20-40K   14      20      40
  7.txt   194    321         8       7 Flash Express 数据分析师 11-22K   16      11      22
  8.txt   175    307         4       8       Shopee 数据分析师 25-50K   15      25      50
  9.txt    85    129         7       9      珀菲克特 数据分析师 30-45K   14      30      45
 10.txt   241    490        17      10      翼鸥教育 数据分析师 25-50K           25      50
 11.txt   217    443         8      11        微博 数据分析师 20-30K   14      20      30
 12.txt   240    477         5      12      京东集团 数据分析师 20-35k   15      20     35k
 13.txt   160    277         8      13    BOSS直聘 数据分析师 25-50K   16      25      50
 14.txt   240    458         8      14        美团 数据分析师 20-27K           20      27
 15.txt   245    449         8      15      便利蜂 数据分析师 30-60K           30      60
```

# 可视化文档变量

```
# 看基本变量
ggplot(data = doc.smy, aes(x = 公司, y = Tokens, group = 1)) + geom_line() +  geom_point()
ggplot(data = doc.s                                                          eom_point()

df<-reshape(doc.smy                                                          idvar = "i
row.names(df)<-NULL
df$time[which(df$ti
df$time[which(df$ti
df$薪数
df$薪数
df$薪数
df
ggplot
  geom
```



# 三种分词方式

```
###############
# 分词
###############
doc.tokens<-tokens(doc.corpus)
doc.tokens

doc.tokens.sentence <- tokens(doc.corpus, what = "sentence")
doc.tokens.sentence[1]

doc.tokens.character <- tokens(doc.corpus, what = "character")
doc.tokens.character[1]
```

按句子分隔

```
> doc.tokens.sentence <- tokens(doc.corpus, what = "sentence")
> doc.tokens.sentence[1]
Tokens consisting of 1 document and 7 docvars.
1.txt :
 [1] "数据分析师 25-50K·14薪  岗位职责：  理解跨境贸易业务，分析跨境资金的行为模式，能设计并实现

 [2] "对异常跨境贸易行为，能用数据指标进行量化，并精确的定位。"

 [3] "能利用统计学、数据挖掘等基础知识，把行为规律模型化。"

 [4] "能够编写文档，把模型思路用文字准确表达。"

 [5] "能够与客户进行沟通。"

 [6] "岗位要求：  3-5年数据分析相关的工作经验，大学本科以上学历。"

 [7] "熟悉R语言，SQL，会用可视化工具。"
```

按词分隔

```
> doc.tokens<-tokens(doc.corpus)
> as.character(doc.tokens[1])
  [1] "数据"     "分析"     "师"       "25-50K"   "."        "14"       "薪"       "岗位"     "职责"
 [10] ":"        "理解"     "跨"       "境"       "贸易"     "业务"     ","        "分析"     "跨"
 [19] "境"       "资金"     "的"       "行为"     "模式"     ","        "能"       "设计"     "并"
 [28] "实现"     "跨"       "境"       "资金"     "监"       "控"       "指标"     "。"       "对"
 [37] "异常"     "跨"       "境"       "贸易"     "行为"     ","        "能"       "用"       "数据"
 [46] "指标"     "进行"     "量化"     ","        "并"       "精确"     "的"       "定位"     "。"
 [55] "能"       "利用"     "统计"     "学"       "、"       "数据"     "挖掘"     "等"       "基础"
 [64] "知识"     ","        "把"       "行为"     "规律"     "模型"     "化"       "。"       "能够"
 [73] "编写"     "文"       "档"       ","        "把"       "模型"     "思路"     "用"       "文字"
 [82] "准确"     "表达"     "。"       "能够"     "与"       "客户"     "进行"     "沟通"     "。"
 [91] "岗位"     "要求"     ":"        "3-5"      "年"       "数据"     "分析"     "相关"     "的"
[100] "工作"     "经验"     ","        "大学"     "本科"     "以上"     "学历"     "。"       "熟悉"
[109] "R"        "语言"     ","        "SQL"      ","        "会"       "用"       "可"       "视"
[118] "化"       "工具"     "。"       "熟悉"     "统计"     "学"       "、"       "数据"     "挖掘"
[127] "的"       "算法"     "模型"     ","        "对"       "模型"     "有"       "自己"     "的"
[136] "理解"     ","        "懂得"     "优"       "化"       "方法"     "。"       "至少"     "熟悉"
[145] "一个"     "行业"     "的"       "一条"     "业务"     "线"       ","        "做"       "过"
[154] "精准"     "分析"     "的"       "业务"     "模型"     "。"       "善于"     "用"       "文字"
```

按字分隔

```
> doc.tokens.character <- tokens(doc.corpus, what = "character")
> as.character(doc.tokens.character[1])
  [1] "数" "据" "分" "析" "师" "2"  "5"  "-"  "5"  "0"  "K"  "·"  "1"  "4"  "薪" "岗" "位" "职" "责" ":"
 [21] "理" "解" "跨" "境" "贸" "易" "业" "务" "，" "分" "析" "跨" "境" "资" "金" "的" "行" "为" "模" "式"
 [41] "，" "能" "设" "计" "并" "实" "现" "跨" "境" "资" "金" "监" "控" "指" "标" "。" "对" "异" "常" "跨"
 [61] "境" "贸" "易" "行" "为" "，" "能" "用" "数" "据" "指" "标" "进" "行" "量" "化" "，" "并" "精" "确"
 [81] "的" "定" "位" "。" "能" "利" "用" "统" "计" "学" "、" "数" "据" "挖" "掘" "等" "基" "础" "知" "识"
[101] "，" "把" "行" "为" "规" "律" "模" "型" "化" "。" "能" "够" "编" "写" "文" "档" "，" "把" "模" "型"
[121] "思" "路" "用" "文" "字" "准" "确" "表" "达" "。" "能" "够" "与" "客" "户" "进" "行" "沟" "通" "。"
[141] "岗" "位" "要" "求" ":" "3"  "-"  "5"  "年" "数" "据" "分" "析" "相" "关" "的" "工" "作" "经" "验"
[161] "，" "大" "学" "本" "科" "以" "上" "学" "历" "。" "熟" "悉" "R"  "语" "言" "，" "S"  "Q"  "L"  "，"
[181] "会" "用" "可" "视" "化" "工" "具" "。" "熟" "悉" "统" "计" "学" "、" "数" "据" "挖" "掘" "的" "算"
[201] "法" "模" "型" "，" "对" "模" "型" "有" "自" "己" "的" "理" "解" "，" "懂" "得" "优" "化" "方" "法"
[221] "。" "至" "少" "熟" "悉" "一" "个" "行" "业" "的" "一" "条" "业" "务" "线" "，" "做" "过" "精" "准"
[241] "分" "析" "的" "业" "务" "模" "型" "。" "善" "于" "用" "文" "字" "和" "语" "言" "进" "行" "表" "达"
[261] "和" "沟" "通" "。" "对" "新" "领" "域" "，" "有" "好" "奇" "心" "，" "能" "主" "动" "学" "习" "新"
```

# 特征清洗和停用词

英文停用词表

```
> stopwords('english')
  [1] "i"          "me"         "my"         "myself"     "we"         "our"        "ours"
  [8] "ourselves"  "you"        "your"       "yours"      "yourself"   "yourselves" "he"
 [15] "him"        "his"        "himself"    "she"        "her"        "hers"       "herself"
 [22] "it"         "its"        "itself"     "they"       "them"       "their"      "theirs"
 [29] "themselves" "what"       "which"      "who"        "whom"       "this"       "that"
 [36] "these"      "those"      "am"         "is"         "are"        "was"        "were"
 [43] "be"         "been"       "being"      "have"       "has"        "had"        "having"
 [50] "do"         "does"       "did"        "doing"      "would"      "should"     "could"
 [57] "ought"      "i'm"        "you're"     "he's"       "she's"      "it's"       "we're"
 [64] "they're"    "i've"       "you've"     "we've"      "they've"    "i'd"        "you'd"
 [71] "he'd"       "she'd"      "we'd"       "they'd"     "i'll"       "you'll"     "he'll"
 [78] "she'll"     "we'll"      "they'll"    "isn't"      "aren't"     "wasn't"     "weren't"
 [85] "hasn't"     "haven't"    "hadn't"     "doesn't"    "don't"      "didn't"     "won't"
 [92] "wouldn't"   "shan't"     "shouldn't"  "can't"      "cannot"     "couldn't"   "mustn't"
 [99] "let's"      "that's"     "who's"      "what's"     "here's"     "there's"    "when's"
[106] "where's"    "why's"      "how's"      "a"          "an"         "the"        "and"
[113] "but"        "if"         "or"         "because"    "as"         "until"      "while"
[120] "of"         "at"         "by"         "for"        "with"       "about"      "against"
[127] "between"    "into"       "through"    "during"     "before"     "after"      "above"
```

中文停用词表

```
> stopwords("zh", source = "misc")
  [1] "按"      "按照"    "俺"      "们"      "阿"      "别"      "别人"    "别处"    "是"
 [10] "别的"    "别管"    "说"      "不"      "不仅"    "不但"    "不光"    "不单"    "不只"
 [19] "不外乎"  "不如"    "不妨"    "不尽"    "然"      "不得"    "不怕"    "不惟"    "不成"
 [28] "不拘"    "料"      "不是"    "不比"    "不然"    "特"      "不独"    "不管"    "不至于"
 [37] "若"      "不论"    "不过"    "不问"    "比"      "方"      "比如"    "及"      "本身"
 [46] "本着"    "本地"    "本人"    "本"      "巴巴"    "巴"      "并"      "并且"    "非"
 [55] "彼"      "时"      "彼此"    "便于"    "把"      "边"      "鄙人"    "罢了"    "被"
 [64] "般"      "的"      "此"      "间"      "此次"    "此时"    "此外"    "处"      "此地"
 [73] "才"      "才能"    "朝"      "朝着"    "从"      "从此"    "从而"    "除非"    "除此之外"
 [82] "除"      "开"      "除外"    "除了"    "诚"      "诚如"    "出来"    "出于"    "曾"
 [91] "趁"      "着"      "在"      "乘"      "冲"      "等等"    "等到"    "等"      "第"
[100] "当"      "当然"    "当地"    "多"      "多么"    "多少"    "对"      "对于"    "对待"
[109] "对方"    "对比"    "得"      "得了"    "打"      "的确"    "的话"    "但"      "但凡"
[118] "但是"    "大家"    "大"      "地"      "待"      "都"      "到"      "叮"      "咚"
[127] "而言"    "而是"    "而已"    "而外"    "而后"    "而"      "况"      "而且"    "尔尔"
[136] "尔"      "后"      "二"      "来"      "独"      "徒"      "非但"    "否则"    "反过来"
[145] "反而"    "反之"    "分别"    "凡是"    "凡"      "个"      "个别"    "固然"    "故"
[154] "故此"    "故而"    "果然"    "果真"    "各"      "各个"    "各位"    "各种"    "各自"
[163] "关于"    "具体"    "归"      "齐"      "根据"    "管"      "赶"      "跟"      "过"
[172] "该"      "给"      "光"      "或者"    "或"      "同"      "或是"    "则"      "何"
```

原始数据

```
> as.character(doc.tokens[1])
  [1] "数据"     "分析"     "师"       "25-50K"   "."        "14"       "薪"       "职位"     "职责"
 [10] ":"        "理解"     "跨"       "境"       "贸易"     "业务"     ","        "及"       "相关"
 [19] "境"       "资金"     "的"       "行为"     "模式"     ","        "能"       "发现"     "异常"
 [28] "实现"     "跨"       "境"       "资金"     "监"       "控"       "指标"     "。"       "对"
 [37] "异常"     "跨"       "境"       "贸易"     "行为"     ","        "能"       "用"       "数据"
 [46] "指标"     "进行"     "量化"     ","        "并"       "精确"     "的"       "定位"     "。"
 [55] "能"       "利用"     "统计"     "学"       "、"       "数据"     "挖掘"     "等"       "基础"
 [64] "知识"     ","        "把"       "行为"     "规律"     "模型"     "化"       "。"       "能够"
 [73] "编写"     "文"       "档"       ","        "把"       "模型"     "思路"     "用"       "文字"
 [82] "准确"     "表达"     "。"       "能够"     "与"       "客户"     "进行"     "沟通"     "。"
 [91] "岗位"     "要求"     ":"        "3-5"      "年"       "数据"     "分析"     "相关"     "的"
[100] "工作"     "经验"     ","        "大学"     "本科"     "以上"     "学历"     "。"       "熟悉"
[109] "R"        "语言"     ","        "SQL"      ","        "会"       "用"       "可"       "视"
[118] "化"       "工具"     "。"       "熟悉"     "统计"     "学"       "、"       "数据"     "挖掘"
[127] "的"       "算法"     "模型"     ","        "对"       "模型"     "有"       "自己"     "的"
[136] "理解"     ","        "懂得"     "优"       "化"       "方法"     "。"       "至少"     "熟悉"
[145] "一个"     "行业"     "的"       "一条"     "业务"     "线"       ","        "做"       "过"
[154] "精准"     "分析"     "的"       "业务"     "模型"     "。"       "善于"     "用"       "文字"
[163] "和"       "语言"     "进行"     "表达"     "和"       "沟通"     "。"       "对"       "新"
[172] "领域"     ","        "有"       "好奇心"   ","        "能"       "主动"     "学习"     "新"
[181] "知识"     "。"       "加分"     "项"       ":"        "有"       "做"       "过量"     "化"
[190] "投资"     "模型"     "熟悉"     "中国"     "经济"     "体系"     "、"       "金融"     "市场"
[199] "熟悉"     "Linux"    "公司"     "介绍"     "北京"     "青"       "萌"       "数"       "海"
[208] "科技"     "有限公司" "是"       "一家"     "以"       "技术"     "为"       "驱动"     "力"
[217] "的"       "数据"     "分析"     "公司"     ","        "为"       "中央"     "政府"     "提供"
[226] "数据"     "分析"     "服务"     ","        "帮助"     "国家"     "级"       "客户"     "提升"
[235] "业务"     "监管"     "和"       "执法"     "能力"     ","        "业务"     "主要"     "涉及"
[244] "国际"     "贸易"     "犯罪"     "、"       "国内"     "反"       "洗钱"     "、"       "反"
[253] "欺诈"     "、"       "行业"     "标准"     "制定"     ","        "经济"     "运行"     "分析"
```

清洗后数据

```
> as.character(t4[1])
  [1] "数据"     "分析"     "师"       "25-50K"   "薪"       "跨"       "境"       "资金"     "行为"
 [10] "分析"     "跨"       "境"       "资金"     "行为"     "模式"     "能"       "发现"     "异常"
 [19] "资金"     "监"       "控"       "指标"     "异常"     "跨"       "境"       "资金"     "行为"
 [28] "数据"     "指标"     "量化"     "精确"     "统计"     "学"       "数据"     "挖掘"     "知识"
 [37] "行为"     "规律"     "模型"     "化"       "文"       "档"       "模型"     "思路"     "文字"
 [46] "准确"     "表达"     "客户"     "沟通"     "3-5"      "数据"     "分析"     "经验"     "大学"
 [55] "本科"     "学历"     "R"        "语言"     "SQL"      "视"       "化"       "工具"     "统计"
 [64] "学"       "数据"     "挖掘"     "算法"     "模型"     "模型"     "优"       "化"       "行业"
 [73] "业务"     "线"       "精准"     "分析"     "业务"     "模型"     "文字"     "语言"     "表达"
 [82] "沟通"     "新"       "领域"     "好奇心"   "主动"     "新"       "知识"     "加分"     "项"
 [91] "过量"     "化"       "投资"     "模型"     "中国"     "经济"     "体系"     "金融"     "市场"
[100] "Linux"    "北京"     "青"       "萌"       "数"       "海"       "有限公司" "技术"     "力"
[109] "数据"     "分析"     "中央"     "政府"     "数据"     "分析"     "服务"     "国家"     "级"
[118] "客户"     "业务"     "监管"     "执法"     "业务"     "国际"     "贸易"     "犯罪"     "国内"
[127] "反"       "洗钱"     "反"       "欺诈"     "行业"     "标准"     "经济"     "运行"     "分析"
[136] "挑战"     "性"       "落地"     "点"       "客户"     "海关"     "总署"     "国家"     "外汇"
[145] "管理局"   "中国"     "证"       "监"       "国家"     "药"       "监"       "局"       "扁平"
[154] "结构"     "环境"     "温和"     "年轻"     "国家"     "法定"     "福利"     "餐"       "补"
[163] "车"       "补"       "国内外"   "团"       "建"       "聪明"     "没有"     "加班"     "基因"
[172] "成长"     "激励"     "政策"
```

# 文字搜索

```
###############
# 文字搜索
###############
kwic(t4, pattern = "算法", valuet
kwic(t4, pattern = "业务", valuet
kwic(t4, pattern = "精通", valuet
kwic(t4, pattern = "管理", valuet
kwic(t4, pattern = "学历", valuet
kwic(t4, pattern = "R|python|sql"
kwic(t4, pattern = "flink|hadoop"
```

```
> kwic(t4, pattern = "R|python|sql", valuetype = "regex")
Keyword-in-context with 38 matches.

   [1.txt, 57]              分析 经验 大学 本科 学历 |      R       | 语言 SQL 视 化 工具
   [1.txt, 59]                 大学 本科 学历 R 语言 |     SQL      | 视 化 工具 统计 学
   [3.txt, 79]               经验 精通 数据 分析 熟练 |     SQL      | Hive Spark Flink Python 脚本
   [3.txt, 81]                数据 分析 熟练 SQL Hive |    Spark     | Flink Python 脚本 语言 熟练
   [3.txt, 83]              熟练 SQL Hive Spark Flink |    Python    | 脚本 语言 熟练 BI 视
   [3.txt, 93]                  视 化 工具 熟练 Excel |     Word     | PPT 办公 软件 强 数据
   [4.txt, 23]                客 分析 用户 价值 渠道 |     ROI      | 线上 渠道 投放 业务 分析
   [4.txt, 69]               适应 数据 分析 数据 期望 |    mysql     | hive spark-sql 查询 语言 shell
   [4.txt, 71]             分析 数据 期望 mysql hive | spark-sql    | 查询 语言 shell 命令 数据
   [4.txt, 79]            shell 命令 数据 期望 scala |    python    | R matlab 同类 工具 中
   [4.txt, 80]          命令 数据 期望 scala python |      R       | matlab 同类 工具 中 数据
   [4.txt, 95]                主 数据 汇 期望 excel |      R       | gnuplot py-matplot py-seaborn echarts tableau
   [4.txt, 98]        期望 excel R gnuplot py-matplot | py-seaborn | echarts tableau 同类 工具 中
   [4.txt, 99] excel R gnuplot py-matplot py-seaborn |   echarts    | tableau 同类 工具 中 模型
  [4.txt, 112]              不限于 多元 拟 合 k-means |      LR      | BDT RF SVM NN 原理
  [4.txt, 114]                  拟 合 k-means LR BDT |      RF      | SVM NN 原理 轻 度
   [5.txt, 91]              分析 计算 背景 数据 SAS |    python    | R 编 程 语言 数据
   [5.txt, 92]            计算 背景 数据 SAS python |      R       | 编 程 语言 数据 数据
   [6.txt, 94]               商 数据 指标 体系 熟练 |     SQL      | 数据 分析 工具 python hadoop
   [6.txt, 98]              熟练 SQL 数据 分析 工具 |    python    | hadoop hive spark 加分 强
  [6.txt, 101]          分析 工具 python hadoop hive |    spark     | 加分 强 独立 强 逻辑
   [7.txt, 54]               学 计算 机 专业 精通 |     SQL      | 数据 业务 经验 沟通 团队
   [7.txt, 78]            数字 敏感 数据 分析 Flash |   Express    | 集团 总部 泰国 曼谷 国内
   [8.txt, 58]                建 模 分析 经验 精通 |     SQL      | BI 工具 Tableau saiku 经验
   [9.txt, 38]              学历 数据 分析 经验 熟练 |     SQL      | EXCEL Python R 统计 机器
   [9.txt, 40]              分析 经验 熟练 SQL EXCEL |    Python    | R 统计 机器 高 数据
   [9.txt, 41]            经验 熟练 SQL EXCEL Python |      R       | 统计 机器 高 数据 敏感度
  [10.txt, 139]              模 经验 经验 Hive 熟练 |     SQL      | 数据 库 查询 语言 强
  [11.txt, 110]              模 分析 项目 经验 熟练 |     SQL      | Hadoop 集群 熟练 hive shell
  [11.txt, 122]            经验 熟练 Excel PPT SAS |      R       | Python 分析 工具 统计 理论
  [11.txt, 123]               熟练 Excel PPT SAS R |    Python    | 分析 工具 统计 理论 统计
   [12.txt, 95]              网 数据 分析 经验 数据 |  hive-SQL    | 数据 提取 工具 熟练 数据
  [12.txt, 103]            工具 熟练 数据 分析 工具 |    python    | R SAS SPSS 思路 逻辑
  [12.txt, 104]          熟练 数据 分析 工具 python |      R       | SAS SPSS 思路 逻辑 性
   [13.txt, 22]                学 数学 专业 编 程 |     sql      | python 机器 经验 业务 思考
   [13.txt, 23]               数学 专业 编 程 sql |    python    | 机器 经验 业务 思考 好奇心
  [14.txt, 125]            学历 三年 数据 分析 运用 |     SQL      | 独立 数据 探查 视 化
  [14.txt, 168]            服务 电子 商务 聚焦 Food |  Platform    | 战略 吃 创新 商 户
```

# 矩阵化

```
###############
# 矩阵化
###############
doc.dfm1<-dfm(doc.tokens)
doc.dfm1

nfeat(doc.dfm1)
featnames(doc.dfm1)

doc.dfm<-dfm(t4)
doc.dfm

nfeat(doc.dfm)
featnames(doc.dfm)

topfeatures(doc.dfm, 200)
```

## 矩阵化

```
> doc.dfm1
Document-feature matrix of: 15 documents, 1,185 features (82.78% sparse) and 7 docvars.
       features
docs    数据 分析 师 25-50k · 14 薪 岗位 职责 :
  1.txt    7    7  1       1 1  1  1    2    1  4
  2.txt   10   10  1       0 1  1  1    2    1  2
  3.txt   14   13  1       0 0  0  0    2    1  0
  4.txt   11    8  2       0 0  0  0    3    0 11
  5.txt   17    4  2       0 1  1  1    0    0  2
  6.txt   12   12  2       0 1  1  1    1    1  0
[ reached max_ndoc ... 9 more documents, reached max_nfeat ... 1,175 more features ]
```

## 主要特征

```
> topfeatures(doc.dfm, 200)
  数据    分析    业务    经验    产品    项目    服务    体系    问题    运营      网      化    逻辑    指标    统计      师
   172     134      87      30      27      22      21      20      20      20      20      19      19      18      18      17
  模型    沟通      商      联    团队      互    需求    用户      学    技术      月    商业      建    专业      强    挖掘
    17      17      17      17      17      16      16      16      15      15      15      14      13      13      13      12
  本科    学历    工具      电    独立    决策    报告    熟练    计算    全球    方案    薪      监      控     sql      线
    12      12      12      12      12      12      12      12      12      11      11      10      10      10      10      10
  集团      直    客户    行业      新    中国      性      机      模    上市    招聘    策略    生活    便利      跨    市场
    10      10       9       9       9       9       9       9       9       9       9       9       9       9       8       8
  国家    专题    数学  python    信息    创新    增长      聘       r    算法    落地    敏感    思维      bi      易    投放
     8       8       8       8       8       8       8       8       7       7       7       7       7       7       7       7
   中      模式    设计    语言      优    领域    经济      点    管理    hive    实施    教育      达   boss    移动     app
     7       6       6       6       6       6       6       6       6       6       6       6       6       6       6       6
   部      合作      京      东      出      蜂      境      项    金融    政府    价值    机会   excel    背景    美国    创业
     6       6       6       6       6       6       5       5       5       5       5       5       5       5       5       5
  城市    安全    海外    渠道    评估    品牌      纳      斯      克    手机    资源    系统    框架    教学      全      博
     5       5       5       5       5       5       5       5       5       5       5       5       5       5       5       5
25-50k    异常      文      视    加分    北京      数    国内      团    流程    购物      日      挂      牌    线上    数字
     4       4       4       4       4       4       4       4       4       4       4       4       4       4       4       4
   构      华      录    社会      客    求职    效率  敏感度    小米    规划      赋    消费      位    解读    东南      亚
     4       4       4       4       4       4       4       4       4       4       4       4       4       4       4       4
  企业      亿      埋      美    便民    贸易    资金    行为      档    思路  好奇心    主动    投资      力      级     ppt
     4       4       4       4       4       3       3       3       3       3       3       3       3       3       3       3
  基金  交易所      家    开发      心    精通    先进    期望      度      主    工程    功能    场景    资产    及时     sas
     3       3       3       3       3       3       3       3       3       3       3       3       3       3       3       3
  销售      业    意识    泰国    研发    大型  shopee      量
     3       3       3       3       3       3       3       3
```

# 分组查询

```
# 按文档变量分组
dfm_group(doc.dfm,groups = 工资)
dfm_group(doc.dfm,groups = 公司)

# 按字典分组
dict <- dictionary(
  list(算法 = c("算法","模型","统计"),
       业务 = c("业务","行业","需求"),
       大数据 = c("hadoop","Spark","Flink"),
       编程 = c("R","Python","Java","sql")))

tokens_lookup(t4,dictionary = dict) %>%
  dfm()%>%
  dfm_group(groups = 工资)

tokens_lookup(t4,dictionary = dict) %>%
  dfm()%>%
  dfm_group(groups = 公司)
```

按文档变量分组

```
> # 按文档变量分组
> dfm_group(doc.dfm,groups = 工资)
Document-feature matrix of: 11 documents, 726 features (81.88% sparse) and 4 docvars.
        features
docs     数据 分析 师 25-50k 薪 跨 境 贸易 业务 资金
  11-22K   10    8  1      0  1  0  0    0    5    0
  14-15K   11    8  2      0  0  0  0    0    2    0
  15-25K   14   13  1      0  0  0  0    0    5    0
  20-27K   19    6  1      0  0  0  0    0    8    0
  20-30K   13   13  1      0  1  0  0    0    8    0
  20-35k   18   16  0      0  1  1  0    0    8    0
[ reached max_ndoc ... 5 more documents, reached max_nfeat ... 716 more features ]
> dfm_group(doc.dfm,groups = 公司)
Document-feature matrix of: 13 documents, 726 features (84.00% sparse) and 2 docvars.
               features
docs            数据 分析 师 25-50k 薪 跨 境 贸易 业务 资金
  BOSS直聘        14   10  3      1  1  0  0    0    4    0
  Flash Express   10    8  1      0  1  0  0    0    5    0
  Shopee          10    8  1      1  1  1  1    0    4    0
  便利蜂           4    6  0      0  0  0  0    0    4    1
  当当网          10   10  1      0  1  1  0    0    6    0
  京东集团        35   20  2      0  2  1  0    0   25    0
[ reached max_ndoc ... 7 more documents, reached max_nfeat ... 716 more features ]
```

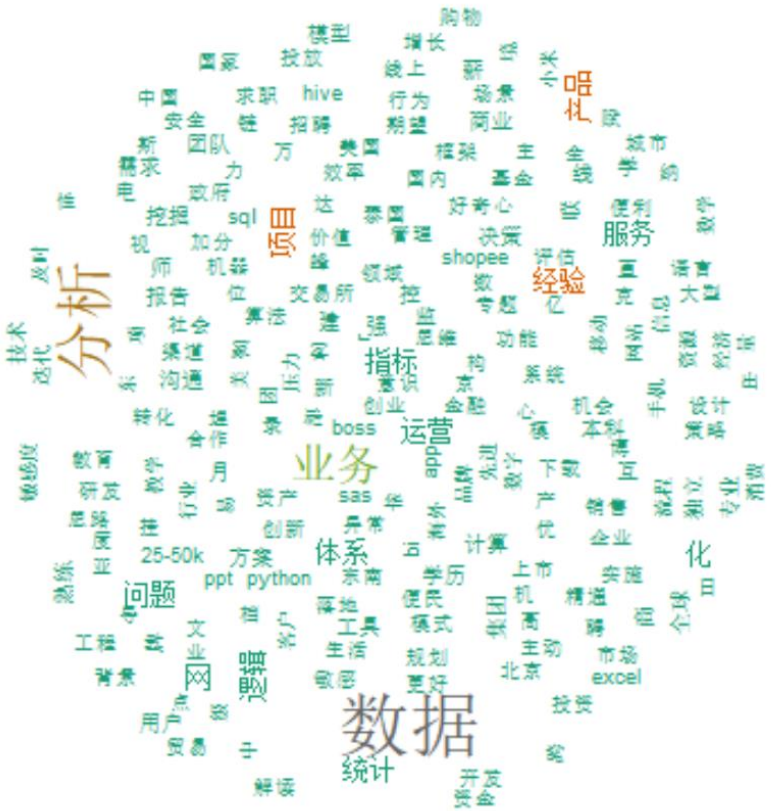
按文本特征分组

```
> tokens_lookup(t4,dictionary = dict) %>%
+   dfm()%>%
+   dfm_group(groups = 工资)
Document-feature matrix of: 11 documents, 4 features (22.73% sparse) and 4 docvars.
        features
docs     算法 业务 大数据 编程
  11-22K    1    5      0    1
  14-15K    3    2      0    3
  15-25K    5   11      2    2
  20-27K    1   11      0    1
  20-30K    7   10      1    3
  20-35k    1    9      0    2
[ reached max_ndoc ... 5 more documents ]
> tokens_lookup(t4,dictionary = dict) %>%
+   dfm()%>%
+   dfm_group(groups = 公司)
Document-feature matrix of: 13 documents, 4 features (23.08% sparse) and 2 docvars.
               features
docs            算法 业务 大数据 编程
  BOSS直聘         4    4      0    5
  Flash Express    1    5      0    1
  Shopee           2    7      0    1
  便利蜂           4    5      0    0
  当当网           1    6      0    0
  京东集团         2   28      0    4
[ reached max_ndoc ... 7 more documents ]
```

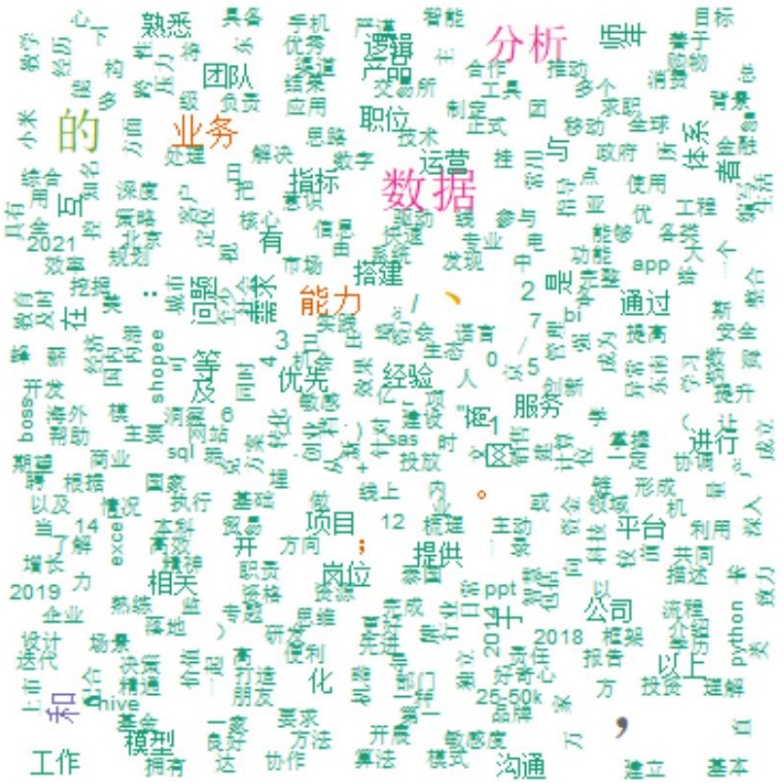
# 词云

```
# 词云
library("quanteda.textplots")
set.seed(1)
textplot_wordcloud(doc.dfm, min_count = 3, random_order=TRUE,
                   rotation = 0.25, min_size = 1, max_size = 4,
                   color = RColorBrewer::brewer.pal(8, "Dark2"))
```

清洗后的数据



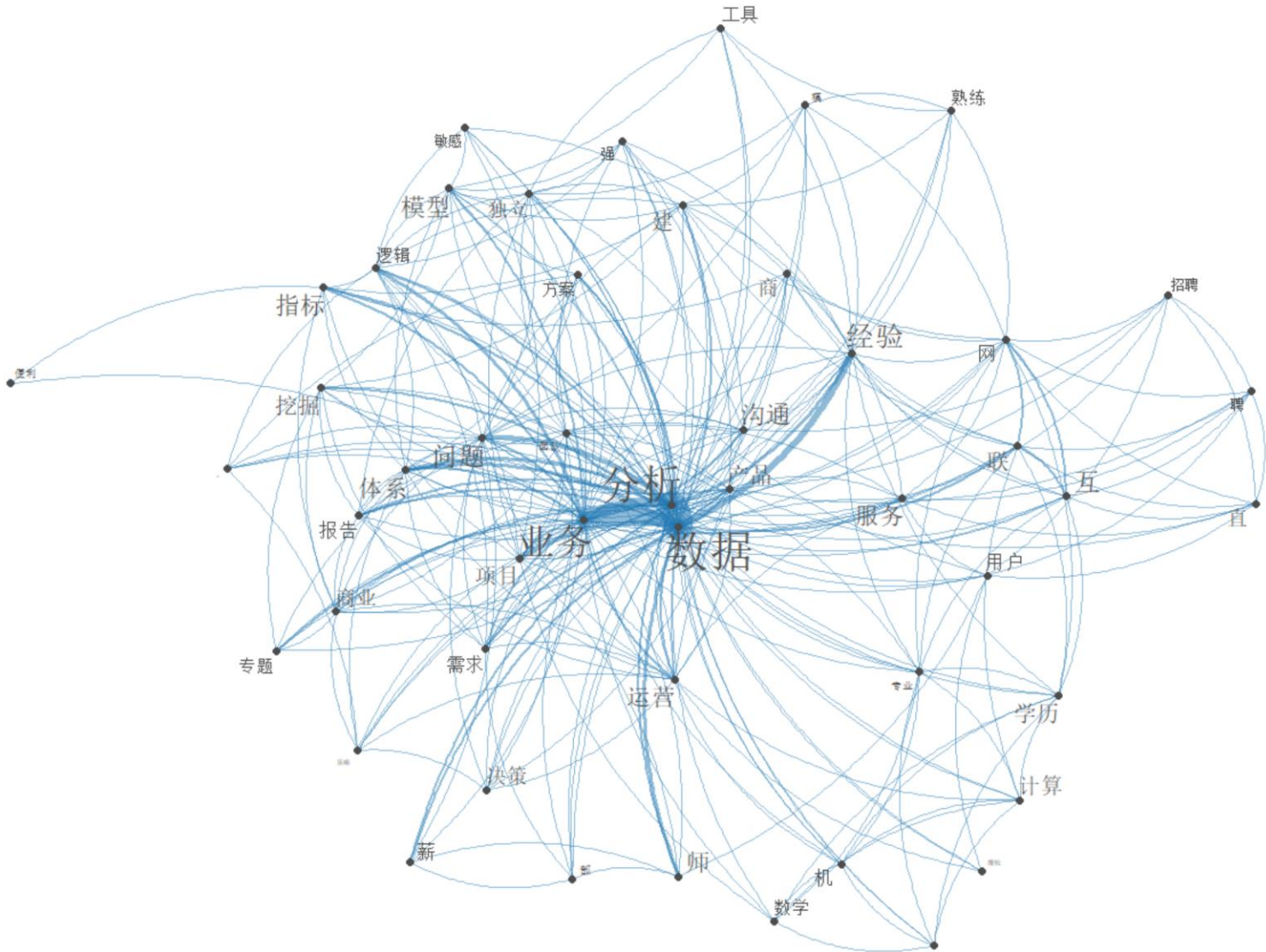

原始数据



# 特征共现矩阵

```
################
# 特征共现矩阵
################
fcmat<-fcm(t4, context = "window", window = 5)
#fcmat<-fcm(t4, context = "document")
#fcmat<-fcm(t4, context = "window", count = "weighted", window =
feat <- names(topfeatures(fcmat, 50));feat
fcmat<-fcm_select(fcmat, pattern = feat)
textplot_network(fcmat,min_freq = 0.5,vertex_labelsize = 1.5 * l
```

```
> feat <- names(topfeatures(fcmat, 50));feat
 [1] "分析" "业务" "数据" "产品" "经验" "逻辑" "团队" "项目"
[11] "问题" "运营" "需求" "专业" "联"   "报告" "熟练" "商业"
[21] "策略" "模型" "决策" "专题" "指标" "方案" "师"   "强"
[31] "沟通" "独立" "机"   "挖掘" "互"   "用户" "增长" "出"
[41] "计算" "工具" "聘"   "招聘" "学历" "数学" "信息" "直"
```

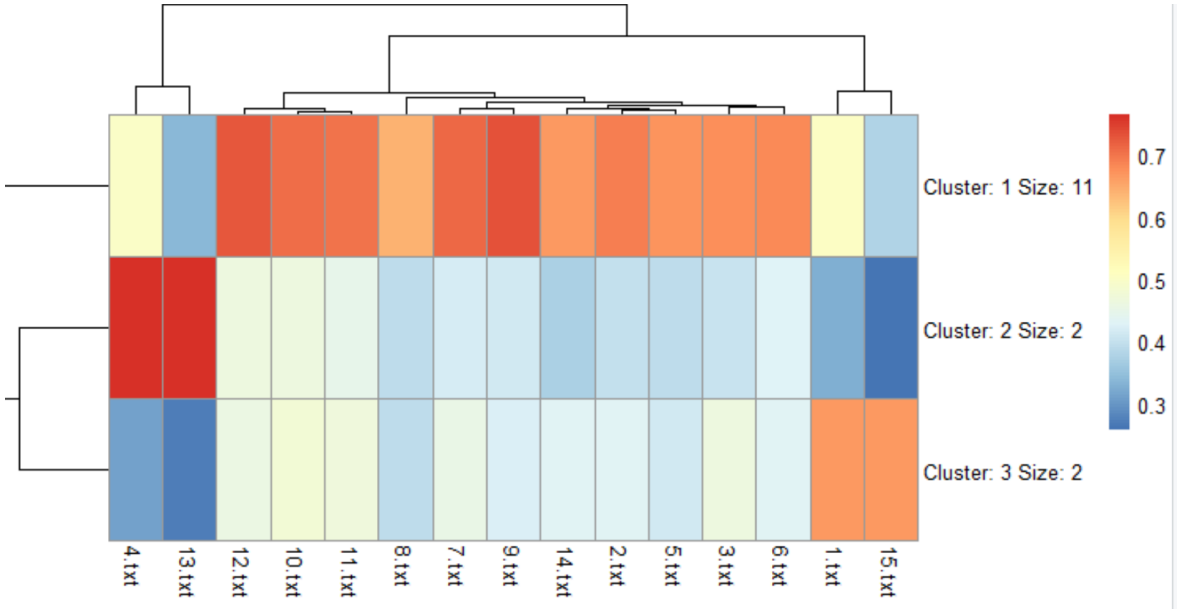


# 文档相似度

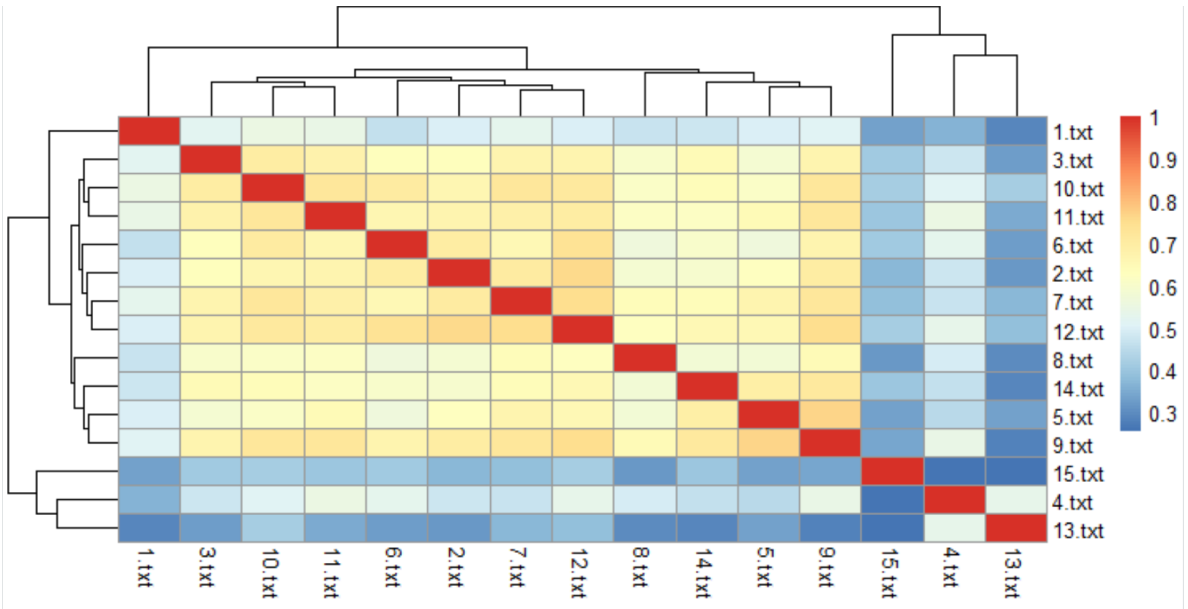
```
# 文本相似
library("quanteda.textstats")
simi <- textstat_simil(doc.dfm,margin = "documents", method = "cosine")
simi

library(pheatmap)
pheatmap(as.matrix(simi))
pheatmap(as.matrix(simi),kmeans_k = 3)
summary(doc.corpus)
```

```
> simi
textstat_simil object; method = "cosine"
       1.txt 2.txt 3.txt 4.txt 5.txt 6.txt 7.txt 8.txt 9.txt 10.txt 11.txt 12.txt 13.txt 14.txt 15.txt
1.txt  1.000 0.495 0.517 0.361 0.496 0.460 0.524 0.471 0.511  0.548  0.541  0.500  0.288  0.474  0.333
2.txt  0.495 1.000 0.630 0.477 0.623 0.695 0.704 0.589 0.696  0.661  0.665  0.756  0.316  0.593  0.371
3.txt  0.517 0.630 1.000 0.478 0.587 0.634 0.669 0.604 0.671  0.699  0.674  0.665  0.324  0.643  0.409
4.txt  0.361 0.477 0.478 1.000 0.448 0.528 0.470 0.486 0.543  0.513  0.546  0.538  0.532  0.457  0.261
5.txt  0.496 0.623 0.587 0.448 1.000 0.567 0.672 0.579 0.762  0.606  0.649  0.653  0.336  0.693  0.333
6.txt  0.460 0.695 0.634 0.528 0.567 1.000 0.657 0.567 0.667  0.702  0.659  0.732  0.329  0.601  0.405
7.txt  0.524 0.704 0.669 0.470 0.672 0.657 1.000 0.637 0.718  0.719  0.687  0.747  0.368  0.639  0.384
8.txt  0.471 0.589 0.604 0.486 0.579 0.567 0.637 1.000 0.650  0.606  0.619  0.627  0.296  0.576  0.317
9.txt  0.511 0.696 0.671 0.543 0.762 0.667 0.718 0.650 1.000  0.718  0.721  0.748  0.285  0.715  0.337
10.txt 0.548 0.661 0.699 0.513 0.606 0.702 0.719 0.606 0.718  1.000  0.718  0.717  0.416  0.637  0.415
11.txt 0.541 0.665 0.674 0.546 0.649 0.659 0.687 0.619 0.721  0.718  1.000  0.695  0.352  0.617  0.399
12.txt 0.500 0.756 0.665 0.538 0.653 0.732 0.747 0.627 0.748  0.717  0.695  1.000  0.389  0.650  0.414
13.txt 0.288 0.316 0.324 0.532 0.336 0.329 0.368 0.296 0.285  0.416  0.352  0.389  1.000  0.289  0.255
14.txt 0.474 0.593 0.643 0.457 0.693 0.601 0.639 0.576 0.715  0.637  0.617  0.650  0.289  1.000  0.397
15.txt 0.333 0.371 0.409 0.261 0.333 0.405 0.384 0.317 0.337  0.415  0.399  0.414  0.255  0.397  1.000
```



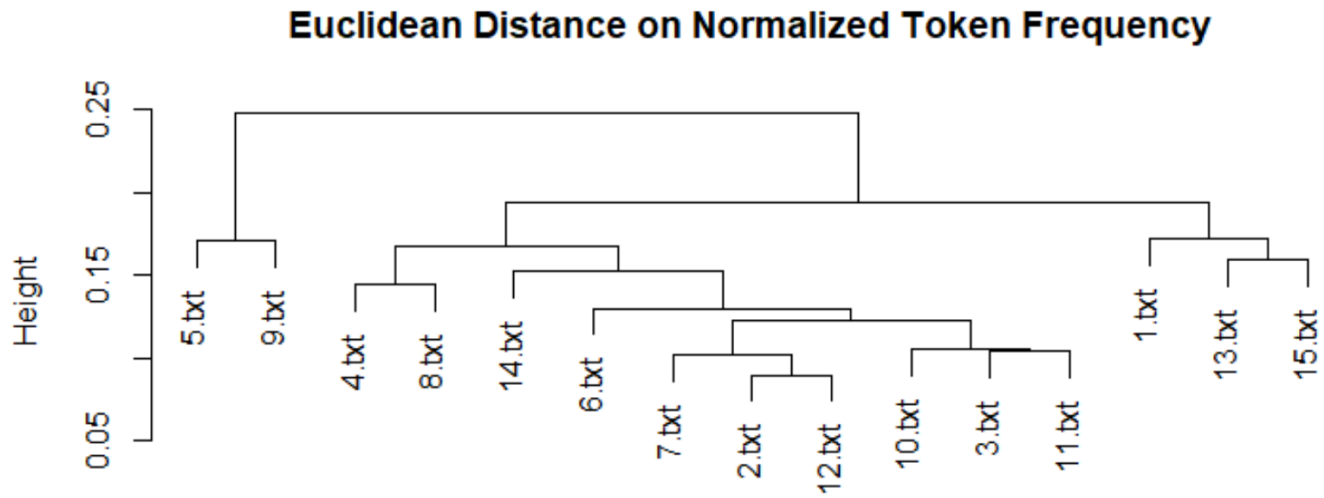

聚类相似度




相似度矩阵

# 文档聚类分析

```r
# 聚类分析
dfm_tmp <- dfm_trim(doc.dfm, min_termfreq = 3, min_docfreq = 3)
#分层聚类 -  在归一化dfm上计算距离
dist_tmp <- textstat_dist(dfm_weight(dfm_tmp, scheme = "prop"))
# 聚类分析文本距离
cluster_tmp <- hclust(as.dist(dist_tmp))
# 按文档名标注
cluster_tmp$labels <- docnames(dfm_tmp)
# 绘制树状图
plot(cluster_tmp, xlab = "", sub = "",
     main = "Euclidean Distance on Normalized Token Frequency")
```



# 词相似性

```
# 词相似
tstat_sim <- textstat_simil(doc.dfm,
                            doc.dfm[, c("数据", "业务", "统计","python","学历")],
                            method = "cosine", margin = "features")
lapply(as.list(tstat_sim), head, 10)
```

```
> lapply(as.list(tstat_sim), head, 10)
$数据
      分析        业务        本科        学历        计算        师        经验        建        机        工具
0.9208772 0.9143359 0.8818895 0.8818895 0.8683605 0.8578983 0.8481793 0.8475083 0.8230244 0.7989704

$业务
      数据        计算        本科        学历        建        经验        沟通        师        分析        薪
0.9143359 0.8762302 0.8688141 0.8688141 0.8593080 0.8411582 0.8347749 0.7847975 0.7718348 0.7589811

$统计
       建         sql        熟练        分析        本科        学历        问题        信息        数据        薪
0.8318903 0.8134892 0.8072074 0.8056200 0.7921180 0.7921180 0.7921180 0.7921180 0.7787037 0.7592566

$python
      数学        数据        信息        师           r        线        效率        熟练        建        分析
0.7500000 0.7175938 0.7144345 0.7071068 0.7071068 0.7071068 0.7071068 0.6933752 0.6859943 0.6773210

$学历
      本科        数据        业务        计算        机        经验        分析        专业        统计        建
1.0000000 0.8818895 0.8688141 0.8660254 0.8660254 0.8498366 0.8484848 0.7947194 0.7921180 0.7701540
```

# 词关联分析

```
# 关联分析
textstat_collocations(t4, size = 2, tolower = TRUE) %>% head(20)
textstat_collocations(t4, size = 2, min_count = 5, tolower = TRUE)%>% head(20)
textstat_collocations(t4, size = 4, tolower = TRUE)%>% head(20)
```

2词关联

```
> textstat_collocations(t4, size = 2, tolower = TRUE) %>% head(20)
    collocation count count_nested length   lambda         z
1     数据 分析    71            0      2 3.235997 16.175443
2       统计 学    10            0      2 6.332405 10.074274
3     指标 体系     8            0      2 5.086008  9.391429
4         监 控     8            0      2 8.137712  8.493181
5      熟练 SQL     5            0      2 6.014403  8.201504
6     专业 本科     5            0      2 5.376820  8.169258
7         建 模     8            0      2 7.859284  7.994207
8      python r     4            0      2 6.577861  7.695874
9     逻辑 思维     5            0      2 5.940951  7.362149
10    用户 增长     4            0      2 5.301313  7.317599
11      学 计算     4            0      2 4.747509  7.263357
12    数学 统计     4            0      2 5.152095  7.187585
> textstat_collocations(t4, size = 2, min_count = 10, tolower = TRUE)
   collocation count count_nested length     lambda         z
1    数据 分析    71            0      2  3.2359969 16.175443
2      统计 学    10            0      2  6.3324054 10.074274
3        联 网    17            0      2 10.1296266  6.617342
4        互 联    16            0      2 10.9192797  6.611469
5     分析 师    16            0      2  5.4030764  6.297439
6        电 商    12            0      2  9.3423649  6.211404
7    分析 报告    10            0      2  4.3904549  6.186266
8    本科 学历    12            0      2 11.7422499  5.812997
9    数据 业务    11            0      2  0.7522626  2.303406
```

4词关联

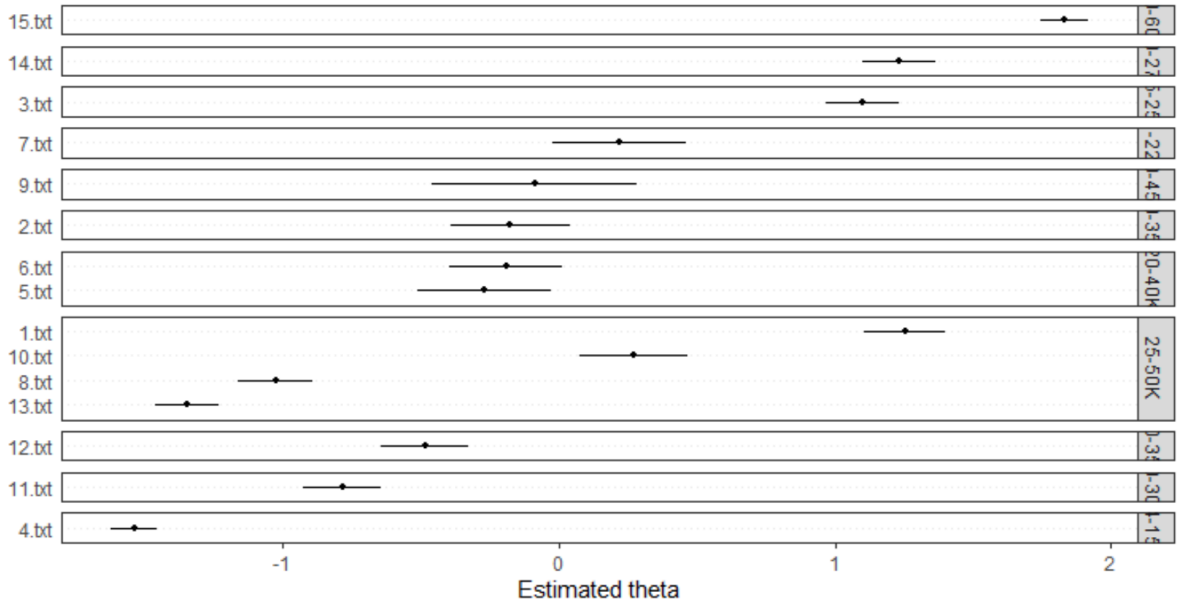
```
> textstat_collocations(t4, size = 5, tolower = TRUE)%>% head(20)
                collocation count count_nested length   lambda         z
1  技术 用户 服务 理念 招聘     2            0      5 6.025990 0.8297966
2    聘 技术 用户 服务 理念     2            0      5 5.935533 0.8172029
3  数据 指标 体系 分析 体系     2            0      5 4.784137 0.7630884
4      落地 客户 服务 全 链     2            0      5 4.674183 0.6496786
5      聘 产品 移动 匹配 直     2            0      5 4.644775 0.6384106
6  思维 业务 解读 独立 领导     2            0      5 4.499075 0.6316092
7  用户 服务 理念 招聘 求职     2            0      5 4.268934 0.5859561
8  规划 数据 指标 体系 分析     2            0      5 3.694887 0.5656072
9    招聘 求职 易 求职 招聘     2            0      5 3.998755 0.5562987
10     模式 线 招聘 APP 月     2            0      5 4.003876 0.5501397
11   求职 易 求职 招聘 BOSS     2            0      5 3.486710 0.4846473
12   服务 理念 招聘 求职 易     2            0      5 3.295852 0.4519102
13     聘 模式 线 招聘 APP     2            0      5 3.270135 0.4489844
14      聘 款 全球 移动 互     2            0      5 2.653949 0.3691086
15    直 聘 技术 用户 服务     2            0      5 2.493268 0.3486200
16      学历 互 联 网 数据     2            0      5 2.422073 0.3479729
17      客户 服务 全 链 路     2            0      5 2.475326 0.3382926
18      互 联 网 数据 分析     5            0      5 2.292701 0.3311241
19     招聘 APP 月 上线 月     2            0      5 2.269683 0.3068753
20    产品 移动 匹配 直 聊     2            0      5 2.077368 0.2806754
```

# Wordfish文档缩放分析

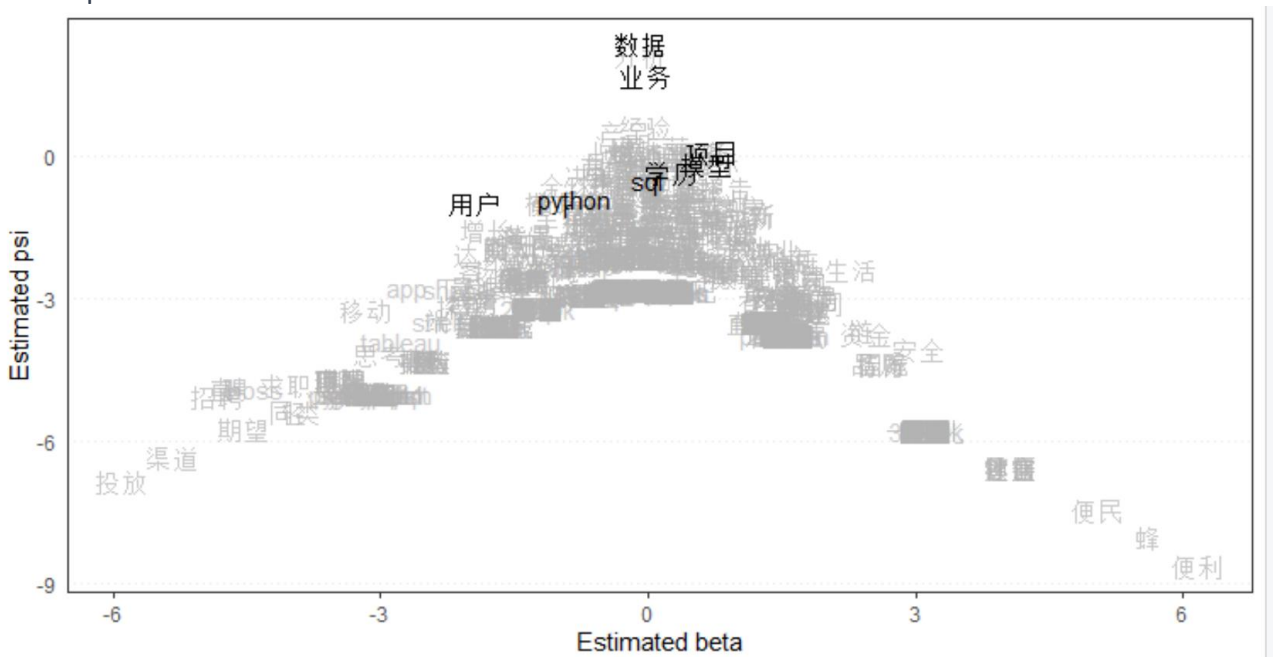
```
# 文档位置：wordfish无监督文档缩放分析
library("quanteda.textmodels")
tmod_wf <- textmodel_wordfish(doc.dfm, dir = c(2, 1))
textplot_scale1d(tmod_wf, groups = docvars(doc.dfm, "工资"))
textplot_scale1d(tmod_wf, groups = docvars(doc.dfm, "公司"))


textplot_scale1d(tmod_wf, margin = "features",
                 highlighted = c("数据", "工资", "学历","python", "r", "模型",
                                 "建模","业务", "sql","跨境","电商","项目","用户"))
```

theta: estimated document positions



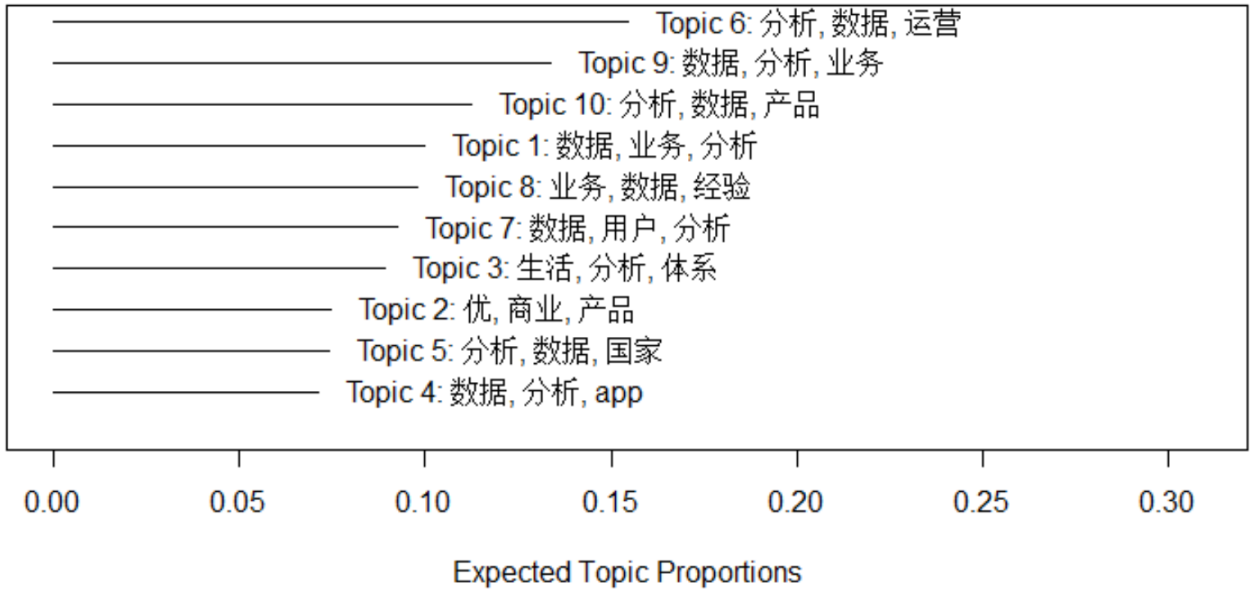

Beta: estimated feature marginal effects
psi: estimated word fixed effects

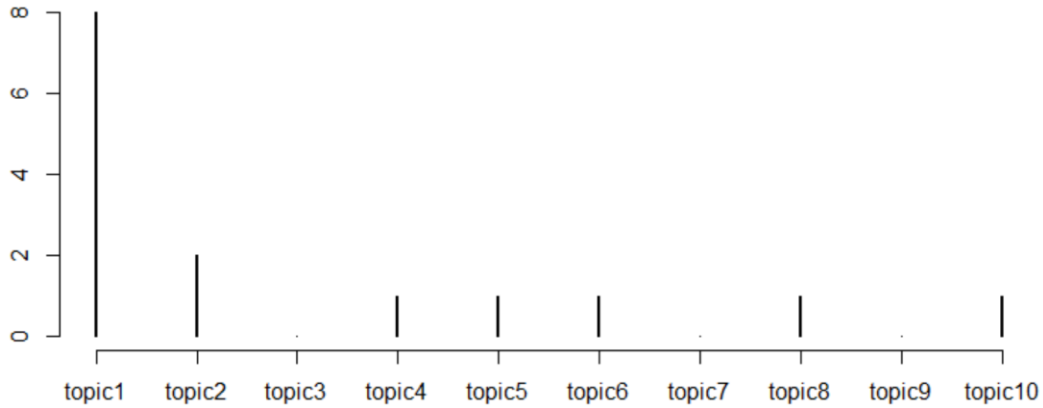


# 主题模型

```
> summary(model_stm)
A topic model with 10 topics, 15 documents and a 156 word dictionary.
Topic 1 Top Words:
         Highest Prob: 数据, 业务, 分析, 逻辑, 项目, 敏感度, 中
         FREX: 敏感度, 中, 出, 逻辑, 构, 数据, 项目
         Lift: 敏感度, 产, 构, 出, 中, 高, 大型
         Score: 敏感度, 中, 出, 产, 机器, 决策, 高
Topic 2 Top Words:
         Highest Prob: 优, 商业, 产品, 化, 数据, 学, 联
         FREX: 优, 商业, 化, 产品, 学, 场景, 易
         Lift: 优, 场景, 25-50k, 商业, 好奇心, 功能, 机器
         Score: 优, 商业, 移动, 场景, 25-50k, 功能, 设计
Topic 3 Top Words:
         Highest Prob: 生活, 分析, 体系, 服务, 指标, 业务, 数据
         FREX: 生活, 创新, 消费, 体系, 业, 品牌, 新
         Lift: 消费, 业, 生活, 品牌, 创新, 日, 系统
         Score: 消费, 生活, 品牌, 创新, 新, 系统, 业
```

```
> terms(model_lda, 10)
      topic1 topic2 topic3 topic4   topic5 topic6 topic7 topic8 topic9 topic10
 [1,] "数据" "项目" "体系" "分析"   "监"   "产品" "数据" "分析" "用户" "数据"
 [2,] "分析" "业务" "指标" "模型"   "客户" "问题" "业务" "本科" "分析" "网"
 [3,] "业务" "经验" "生活" "化"     "跨"   "团队" "统计" "中国" "增长" "联"
 [4,] "运营" "计算" "全球" "服务"   "化"   "模"   "师"   "机"   "商"   "商业"
 [5,] "决策" "逻辑" "强"   "优"     "经验" "专业" "学历" "需求" "移动" "互"
 [6,] "商"   "实施" "服务" "建"     "经济" "学"   "逻辑" "集团" "app"  "月"
 [7,] "专题" "信息" "性"   "团队"   "国家" "出"   "挖掘" "bi"   "师"   "沟通"
 [8,] "上市" "服务" "创新" "25-50k" "领域" "r"    "经验" "指标" "经验" "数学"
 [9,] "熟练" "方案" "业务" "设计"   "技术" "新"   "薪"   "金融" "工具" "达"
[10,] "团队" "全"   "品牌" "sql"    "熟练" "全球" "沟通" "点"   "算法" "模式"
```



# 公司介绍

**北京青萌数海科技有限公司** 是一家以技术为驱动力的数据分析公司，为中央政府提供数据分析服务，帮助国家级客户提升业务监管和执法能力，业务主要涉及贸易犯罪、反洗钱、反欺诈、行业标准制定，经济运行分析等，公司长期服务于海关总署，国家外汇管理局，国家药监局，中国版权保护中心等机构。

专注于探索未知领域，数据与业务相结合，可解释，可落地的数据分析。

微信公众号

数据分析师职位