# 漫谈ES中的分词与检索

作者:Ansj　微博:http://weibo.com/ansjsun/　site:http://www.ansj.org

# 中文分词

- 汉语自动分词是任何中文自然语言处理系统都难以回避的第一道基本“工序”，其作用是怎么估计都不会过分。只有逾越这个障碍，中文处理系统才称得上初步打上了 “智能”的印记，构建于词平面之上的各种后续语言分析手段才有展示身手的舞台。否则，系统便只能被束缚在字平面上。

- 分词在很多现实应用领域（中文文本的自动检索、过滤、分类及摘要，中文文本的自动校对，汉外机器翻译，汉字识别与汉语语音识别的后处理，汉语语音合成，以句子为单位的汉字键盘输入，汉字简繁体转换等）中都扮演着极为重要的角色

# 中文歧义的识别

- 交叉歧义（多种切分交织在一起）：内塔内亚胡/说的/确实/在理
- 组合歧义（不同情况下切分不同）：这个人/手上有痣、我们公司人手不足
- 真歧义（几种切分都可以）：乒乓球拍/卖/完了、乒乓球/拍卖/完了

# 实体名识别

- 人名识别:人名特征词(姓氏,常用人名字)+上下文
- 机构名识别:(和人名识别差不多) 难度最大,主要因为词长不固定
- 专有名词:确定文本分类.利用crf或者规则的方式来识别.不同行业.准确率不一
- 网络新词:最好是基于词典.词的无边界线.注定无法穷举,但是能解决常用词就解决了80%的问题.难点是时效性高.需要考虑基于网络文本热点词语发现

# 新词热词发现

- 目前常用的新词发现还是一个比较有研究性的课题，虽然有些论文在准确率很高，但是大多是封闭测试，这意味着结果很难应用到实际工程中。目前Ansj采用的新词发现方式比较简单，采用了高频词的匹配方式，不使用规则，用统计重复串识别新词，根据词性去掉干扰词，虽然有一定的效果，但还是差强人意。
- 统计为主+规则过滤+内部凝固程,词外部合理
- Example: 张三是一个屌丝一样的人物
  - 李四的屌丝毫不比张三差

# 颗粒度问题

- 最难的问题.颗粒度的大小很难把握
- 颗粒度越小歧义越高,歧义多召回率高
- 颗粒度越大,准确率越低,召回率低
- 系统不同需要的平衡点不同
- Example: 中国银行知春路分行
  - NLP语法分析:中国银行/知春路分行
  - 搜索:[中国/银行][中国银行][知春路/分行][知春路分行]

# 一些有趣的case

- 他说的确实在理
- 结婚的和尚未结婚的
- 上海大学城书店
- 北京大学生前来应聘
- 学习近平和李克强将成为一种风尚
- 发展中国家庭养猪事业
- 门把手坏了,门把手夹了
- 两毛五一斤.一斤八两
- 一次性交多少钱

# 一些有趣的case—胡搅蛮缠

- 1. 冬天：能穿多少穿多少； 夏天：能穿多少穿多少。
- 2. 剩女产生的原因有二，一是谁都看不上，二是谁都看不上。
- 3.单身人的来由：原来是喜欢一个人，现在是喜欢一个人。
- 4.两种人容易被甩：一种不知道什么叫做爱，一种不知道什么叫做爱。

# 一些有趣的case

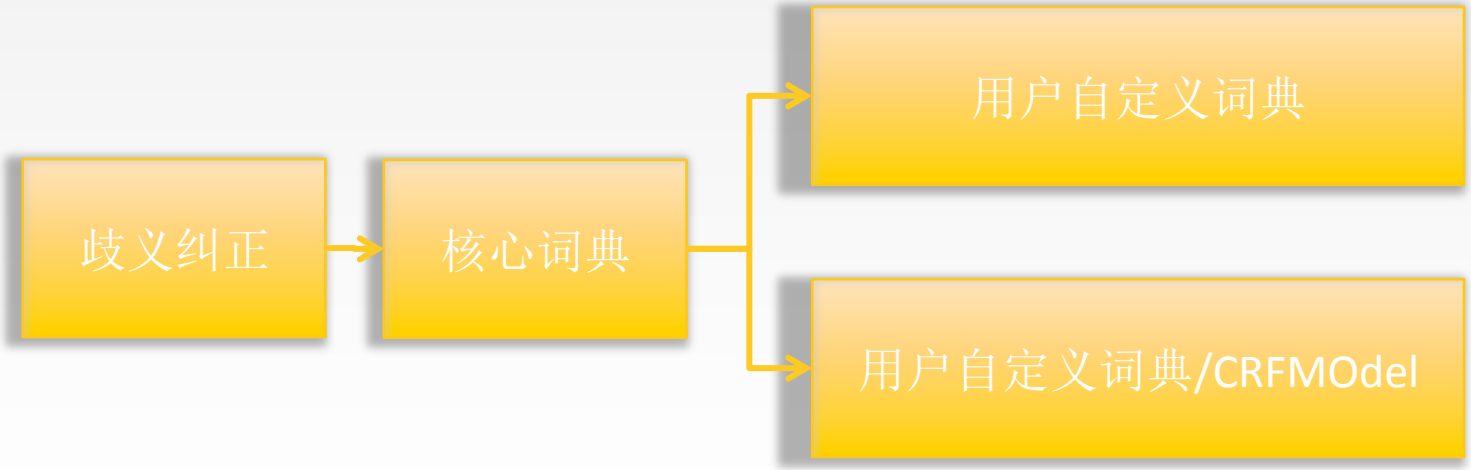
- 研表究明，汉字的序顺并不定一能影阅响读，比如当你看完这句话后，才发这现里的字全是都乱的。
- 人类阅读.视觉窗口.与字体大小无关,大脑的自动补全纠错机制和计算机完全不同.
- 语言学和计算机语言学的差异
- 我们比赛(快跑/慢跑) 语言学是正确的.具有先验只是
- 我们比赛(跳高/跳低) 语言学say no. 计算机语言学 yes 未来性,抛弃逻辑性


# Ansj中文分词

- 多种分词调用方式(BaseAnalysis,ToAnalysis,NlpAnalysis,IndexAnalysis)
- 用户自定义词典
- 歧义纠正词典
- http接口
- 新词发现
- 停用词

# Ansj中文分词

- 分词顺序(基本分词/NLP分词)




# Ansj中文分词-番外篇

- Forest 用户自定义词典的灵魂
- 添加删除用户自定义词典
- 多词典的用法
- 歧义词典的用法

# 分词和搜索的关系

- 搜索的数据结构(成熟稳定)
- 分词可以让搜索:
- 更加智能,和语义相关 (旅游和服务是一流的)
- 减少索引大小.让倒排链更加均匀
- 可以利用词频.做相关度排序.提高排序准确率
- 对于近义词和同义词.以及词的权重可以获得很好的控制
- 遗憾的是也许没有分词.可以让我们过的更好

# 分词badcase


oschina.net
和服
搜 索
所有
软件
讨论
博客
代码
翻译
资讯
所有分类
所有子类
所有编程语言
所有操作系统
只搜索软件名
筛选
找到约 212 条结果 （用时约 0.03 秒） 当前第 1 页，共 11 页
应用发布和服务系统 ShipBuilder
http://www.oschina.net/p/shipbuilder
ShipBuilder 是一个基于 Git 的应用发布和服务系统，使用 Go 语言编写。 主要组件： ShipBuilder 命令行客户端 ShipBui
容器管理 (LXC
网络和服务器监控系统 CactiFans
http://www.oschina.net/p/cactifans
直观看到nagios的状态，可对各种网络设备和服务器进行监控。...
Node.js 的 SOAP 客户端和服务器 Node-SOAP
http://www.oschina.net/p/node-soap
Node-SOAP 是基于 Node.js 的 SOAP 客户端和服务器。 该模块可以让你使用SOAP连接到Web服务。它还提供了一个服务
你运行你自己的SOAP服务。 特性 非常简单

# 分词badcase

全部 讨论 博客 新闻 专栏 群组 会员

站内高级搜索 孙健 高级搜索 帮助

全部 讨论 博客 新闻 专栏 群组 用户

**讨论** MORE

**[世预赛][国足]中国vs卡塔尔**

rt 预计首发 引用门将：宗磊 后卫：孙祥、李玮峰、徐云龙、张帅 中场：朱挺、周海滨、郑智、孙继海、蒿俊闵 前锋：韩鹏 输了的话，2010南非byebye

neusun 发表时间: 2008-06-02 浏览 (1128) 回复 (8) 相关度: 100.00 %

**中国 0 : 1 卡塔尔**

孙继海在场外被红牌 杜伊表现很平静

Feiing 发表时间: 2008-06-07 浏览 (2039) 回复 (12) 相关度: 51.50 %

**春晚美女**

在观众席上发现的 穿蓝色军装的mm 不知道是哪个文工团的。。。。

seen 发表时间: 2009-01-26 浏览 (6147) 回复 (23) 相关度: 47.59 %

**除夕夜中国vs伊拉克**

预祝平局一场 2：2 郑智回归！ ---------- 中国队预计首发(5-3-2)： 门将：宗磊 后卫：孙继海、李玮峰、冯潇霆、徐云龙、张帅 中场：李彦、郑智、刘健 前锋：朱挺、曲波 替补队员：陈东、董方卓、王栋、孙祥、杜震宇、周海滨、张永海 另：4-5-1 门将：宗磊 后卫：孙祥、李玮峰、徐云龙、张帅 中场：李彦、郑智、周海滨，杜震宇，王栋 前锋：曲波

neusun 发表时间: 2008-02-04 浏览 (7019) 回复 (64) 相关度: 39.12 %

# 分词badcase

我经济南下车到广州

一下

广州南站到广州北站 我在容桂做轻轨去到广州南 要去北..._百度知道

3个回答 - 提问时间: 2011年05月21日

广州南站到广州北站 我在容桂做轻轨去到广州南 要去北站接人 是直接做高铁还是坐公车经济一点票价差不多。从广州南到广州北二等票价是22块,跟坐公交差不多,...

zhidao.baidu.com/link?... 2011-05-21 - 百度快照 - 87%好评

从广州经济开发区东区到南站怎么坐车啊?_百度知道

1个回答 - 提问时间: 2013年10月14日

最佳答案: 1、你可以坐B31到文冲市场站落步行到文冲地铁站坐5号线到广州火车站转2号线到广州南站落2、坐B31到BRT夏园站转B1到BRT黄村站转B7快线到洛溪桥脚站...

zhidao.baidu.com/link?... 2013-10-16 - 百度快照 - 87%好评

我在广州经济开发区怎么到广州火车南站 - 已回答 - 搜搜问问

我来回答 匿名 回答(1) 高飞 16级 2010-01-30 广州南站运营后,近期将... 起点广州经济开发区步行至 大沙头码头 乘坐229路 在珠影下车 步行至 ...

wenwen.soso.com/z/q177... 2010-01-30 - 百度快照 - 评价

# 分词badcase


新闻 网页 贴吧 知道 音乐 图片 视频 地图 文库 更多»
送亲qq
QQ空间关闭申请
我要提意见 我要关闭QQ空间 Copyright 1998 - 2013 Tencent. All Rights Reserved. 腾讯公司 版权所有 ...
ctc.qzs.qq.com/qzone/web/loa...htm 2012-12-26 - 百度快照
怎么关闭QQ 百度知道
4个回答 - 提问时间: 2011年02月25日
最佳答案: 关闭/注销QQ空间的申请地址: http://service.qq.com/info/2052.html 在关闭前,QQ会让你填写关闭问卷调查,然后对问题结果进行审核,审核通过后即可...
zhidao.baidu.com/question/2294472... 2011-3-21
QQ空间如何关。 4个回答 2010-01-17
怎么关闭QQ空间,我现在不想要QQ空间了,不知道该怎么关掉。 2个回答 2009-07-18
QQ空间怎样关闭? 6个回答 2009-03-02
更多知道相关问题>>
@径舟-百度
weibo.com/u/2156428565

# ElasticSearch

- 基于lucene,超越lucene。
- 海量规模数据实时分析
- 近似于数据库的聚合功能
- 并非是一个全文检索系统.蜕变为一个完整的数据分析平台
- 典型用户


# Elasticsearch-sql

- 起因:es的查询语句比较反人类,api方式侵入性太高.需要导入几十m的jar包.而且api的硬编码方式不够灵活.相对学习成本较高.
- 解决: 利用sql的方式来查询elasticsearch的索引.利用es中的新特性拓展sql结果
- 项目地址插件安装方式:https://github.com/NLPchina/elasticsearch-sql

# Elasticsearch-sql-一般查询

- https://github.com/NLPchina/elasticsearch-sql/blob/master/src/test/java/org/nlpcn/es4sql/QueryTest.java
- 复杂的boolean类型查询,嵌套查询,一般用在做日志处理方面
- select * from bank where (gender='m' and (age> 25 or account_number>5)) or (gender='w' and (age>30 or account_number < 8)) and email is not miss order by age,_score desc limit 10
- { "from" : 0, "size" : 10, "query" : { "bool" : { "must" : { "bool" : { "should" : [ { "bool" : { "must" : { "term" : { "gender" : "m" } }, "should" : [ { "range" : { "age" : { "from" : 25, "to" : null, "include_lower" : false, "include_upper" : true } } }, { "range" : { "account_number" : { "from" : 5, "to" : null, "include_lower" : false, "include_upper" : true } } } ] } }, { "bool" : { "must" : [ { "term" : { "gender" : "w" } }, { "bool" : { "must_not" : { "filtered" : { "query" : { "match_all" : { } }, "filter" : { "missing" : { "field" : "email" } } } } } } ], "should" : [ { "range" : { "age" : { "from" : 30, "to" : null, "include_lower" : false, "include_upper" : true } } }, { "range" : { "account_number" : { "from" : null, "to" : 8, "include_lower" : true, "include_upper" : false } } } ] } } ] } } }, "sort" : [ { "age" : { "order" : "asc" } }, { "_score" : { "order" : "desc" } } ]}

# Elasticsearch-sql-检索查询-luceneDSL

- https://github.com/NLPchina/elasticsearch-sql/blob/master/src/test/java/org/nlpcn/es4sql/MethodQueryTest.java
- 类似于sql中的like但是索引结构是基于term来搜索的.可以简单认为等同于lucene查询拼入了sql表达式中.可以做全文检索等工作
- luceneDSL:select * from bank where q= query('address:880 Holmes Lane')


# Elasticsearch-sql-检索查询-普通查询

- matchQuery:select * from bank where address= matchQuery('880 Holmes Lane’)
- { "match" : { "address" :* {"query":"880 Holmes Lane", "type" : "boolean" } } }

# Elasticsearch-sql-检索查询-分数设定

- scoreQuery:  select address from bank where address= score(matchQuery('Lane'),100) or address= score(matchQuery('Street'),0.5) order by _score desc limit 3
- "query" : { "bool" : { "must" : { "bool" : {"should" : [ { "constant_score" : { "query" : { "match" : { "address" : {"query" : "Lane", "type" : "boolean" } } }, "boost" : 100.0 } }, {"constant_score" : { "query" : { "match" : { "address" : { "query" :"Street", "type" : "boolean" } } }, "boost" : 0.5 } } ] } } } }


# Elasticsearch-sql-检索查询-通配符

- wildcardQuery: select address from bank where address= wildcardQuery('l*e')  order by _score desc limit 3
- "wildcard": { "address" : { "wildcard" : "l*e" } }


# Elasticsearch-sql-检索查询-精确匹配

- wildcardQuery: select address from bank where address= matchPhrase('671 Bristol Street')  order by _score desc limit 3
- "address" : {"query" : "671 Bristol Street”,"type" : "phrase”}


# Elasticsearch-sql-聚合查询

- https://github.com/NLPchina/elasticsearch-sql/blob/master/src/test/java/org/nlpcn/es4sql/AggregationTest.java
- 实现了group by,count ,sum avg ,max ,min , topHits,count(distinct) 等聚合函数的查询

# Elasticsearch-sql-聚合查询-sum count avg

- Sum count avg
- select sum(age),count(*), count(distinct age) from bank group by gender order by count(distinct age) desc limit 3
- { "size" : 0, "aggregations" : { "gender" : { "terms" : { "field" : "gender", "size" : 3, "order" : { "_count" : "desc" } }, "aggregations" : { "SUM(age)" : { "sum" : { "field" : "age" } }, "COUNT(*)" : { "value_count" : { } }, "COUNT(DISTINCT age)" : { "cardinality" : { "field" : "age" } } } } } }

# Elasticsearch-sql-聚合查询-别名设置

- 别名
- select sum(age),count(*) as kk, count(age) as k from bank  group by gender order by kk asc limit 10
- { "size" : 0, "aggregations" : { "gender" : { "terms" : { "field" : "gender", "size" : 10, "order" : { "_count" : "asc" } }, "aggregations" : { "SUM(age)" : { "sum" : { "field" : "age" } }, "kk" : { "value_count" : { } }, "k" : { "value_count" : { "field" : "age" } } } }


# Elasticsearch-sql-聚合查询-min max

- 其他聚合函数
- select min(age) from bank  group by gender
- select max(age) from bank  group by gender


# Elasticsearch-sql-聚合查询-group

- 区段group 聚合
- select count(age) from bank  group by range(age, 20,25,30,35,40)
- Age按照大于等于20小于25，大于等于25小于30….聚合

# Elasticsearch-sql-聚合查询-group

- 区段group 聚合
- select insert_time from online  group by date_histogram(field='insert_time','interval'='1.5h','format'='yyyy-MM')
- Insert_time 按照 1.5h 一个区段进行聚合.format是key的格式
- select online from online  group by date_range(field='insert_time','format'='yyyy-MM-dd' ,'2014-08-18','2014-08-17','now-8d','now-7d','now-6d','now')
- 支持now关键字,不解释了.

# Elasticsearch-sql-聚合查询-tophits

- 区段group 聚合
- select topHits('size'=3,age='desc') from bank/type  group by gender
- 非二维表.需要看文档
- http://www.elasticsearch.org/guide/en/elasticsearch/reference/current/search-aggregations-metrics-top-hits-aggregation.html


# 结束