# 中文自然语言处理相关整理

## 1.相关算法

### 1.1基于词典分词算法

也称字符串匹配分词算法。该算法是按照一定的策略将待匹配的字符串和一个已建立好的“充分大的”词典中的词进行匹配，若找到某个词条，则说明匹配成功，识别了该词。常见的基于词典的分词算法分为以下几种：**正向最大匹配法、逆向最大匹配法**和**双向匹配分词法**等。

基于词典的分词算法是应用最广泛、分词速度最快的。很长一段时间内研究者都在对基于字符串匹配方法进行优化，比如最大长度设定、字符串存储和查找方式以及对于词表的组织结构，比如采用TRIE索引树、哈希索引等。

### 1.2基于统计的机器学习算法

这类目前常用的是算法是**HMM、CRF、SVM、深度学习**等算法，比如stanford、Hanlp分词工具是基于CRF算法。以CRF为例，基本思路是对汉字进行标注训练，不仅考虑了词语出现的频率，还考虑上下文，具备较好的学习能力，因此其对歧义词和未登录词的识别都具有良好的效果。

Nianwen Xue在其论文《Combining Classifiers for Chinese Word Segmentation》中首次提出对每个字符进行标注，通过机器学习算法训练分类器进行分词，在论文《Chinese word segmentation as character tagging》中较为详细地阐述了基于字标注的分词法。

常见的分词器都是使用**机器学习算法和词典相结合，一方面能够提高分词准确率，另一方面能够改善领域适应性。**

随着深度学习的兴起，也出现了**基于神经网络的分词器**，例如有人员尝试使用双向LSTM+CRF实现分词器，**其本质上是序列标注**，所以有通用性，命名实体识别等都可以使用该模型，据报道其分词器字符准确率可高达97.5%。算法框架的思路与论文《Neural Architectures for Named Entity Recognition》类似，利用该框架可以实现中文分词。

## 2.分词工具

中科院计算所NLPIR [http://ictclas.nlpir.org/nlpir/](https://link.zhihu.com/?target=http%3A//ictclas.nlpir.org/nlpir/)

ansj分词器 [https://github.com/NLPchina/ansj\_seg](https://link.zhihu.com/?target=https%3A//github.com/NLPchina/ansj_seg)

哈工大的LTP [https://github.com/HIT-SCIR/ltp](https://link.zhihu.com/?target=https%3A//github.com/HIT-SCIR/ltp)

清华大学THULAC [https://github.com/thunlp/THULAC](https://link.zhihu.com/?target=https%3A//github.com/thunlp/THULAC)

斯坦福分词器 [https://nlp.stanford.edu/software/segmenter.shtml](https://link.zhihu.com/?target=https%3A//nlp.stanford.edu/software/segmenter.shtml)

Hanlp分词器 [https://github.com/hankcs/HanLP](https://link.zhihu.com/?target=https%3A//github.com/hankcs/HanLP)

结巴分词 [https://github.com/yanyiwu/cppjieba](https://link.zhihu.com/?target=https%3A//github.com/yanyiwu/cppjieba)

KCWS分词器(字嵌入+Bi-LSTM+CRF) [https://github.com/koth/kcws](https://link.zhihu.com/?target=https%3A//github.com/koth/kcws)

ZPar [https://github.com/frcchang/zpar/releases](https://link.zhihu.com/?target=https%3A//github.com/frcchang/zpar/releases)

IKAnalyzer [https://github.com/wks/ik-analyzer](https://link.zhihu.com/?target=https%3A//github.com/wks/ik-analyzer)

**以及部分分词器的简单说明：**

**哈工大的分词器**：主页上给过调用接口，每秒请求的次数有限制。

**清华大学THULAC**：目前已经有Java、Python和C++版本，并且代码开源。

**斯坦福分词器**：作为众多斯坦福自然语言处理中的一个包，目前最新版本3.7.0， Java实现的CRF算法。可以直接使用训练好的模型，也提供训练模型接口。

**Hanlp分词**：求解的是最短路径。优点：开源、有人维护、可以解答。原始模型用的训练语料是人民日报的语料。可用自己的语料进行训练。

**结巴分词工具**：基于前缀词典实现高效的词图扫描，生成句子中汉字所有可能成词情况所构成的有向无环图 (DAG)；采用了动态规划查找最大概率路径, 找出基于词频的最大切分组合；对于未登录词，采用了基于汉字成词能力的 HMM 模型，使用了 Viterbi 算法。

**字嵌入+Bi-LSTM+CRF分词器**：本质上是序列标注，这个分词器用人民日报的80万语料，据说按照字符正确率评估标准能达到97.5%的准确率。

**ZPar分词器**：新加坡科技设计大学开发的中文分词器，包括分词、词性标注和Parser，支持多语言，据说效果是公开的分词器中最好的，C++语言编写。

## 3.分词效果测试

开源项目 [https://github.com/ysc/cws\_evaluation](https://link.zhihu.com/?target=https%3A//github.com/ysc/cws_evaluation) 对多款分词器速度和效果进行过对比。可供参考。

## 4.实现思路以及可能需要的资源

1.使用开源模型，自定义词典文件

例如：使用jieba分词工具，自定词典格式为一个词占一行；每一行分三部分：词语、词频（可省略）、词性（可省略），用空格隔开，顺序不可颠倒。

创新办 3 i

云计算 5

凱特琳 nz

为了在法律内容上达到较好的分词效果，可以自定义法律词典文件，引入相应的中文自然语言处理工具中。

2.使用相关算法，训练自己的模型

例如：开源项目FoolNLTK，基于BiLSTM模型训练而成。使用时可以提供自己的CRF训练集，训练得到自己的模型。

## 5.法律语料库

网络上未找到免费资源，需要华师提供。