基于动态规划的中文分词算法

作者：张昊　　学号：1600012439　　指导老师：陈斌

【摘要】在不以空格作为单词界限的汉语中，分词技术是自然语言处理中的一个非常基础而又关键的环节。在日常生活中，中文分词被广泛应用于中文搜索引擎、计算机系统的汉语接口、机器翻译、汉语语言理解等很多领域。本文从中文分词的研究现状出发,首先列举了一些具有代表性的典型分词系统,比较了当今主流的三种分词方法:基于字符串匹配、基于理解和基于统计的分词方法,并介绍了分词问题中的歧义和未登录词识别两大难点，并对基于动态规划的分词算法的发展方向进行展望。

【关键词】 中文分词；动态规划；算法优化；发展方向

---

0 引言

众所周知，英文是以词为单位的，词和词之间是靠空格隔开，计算机可以很简单通过空格知道一个单词。但在中文中这就复杂许多，同样一句话可以有不同分法，如“合肥一中/学生”和“合肥/一/中学生”。把中文的汉字序列切分成有意义的词，就是中文分词，也称为切词。对于一句话，人可以通过自己的知识来明白哪些是词，哪些不是词，但如何让计算机也能理解？其处理过程就是分词算法。

现有的分词算法可分为三大类：基于字符串匹配的分词方法、基于理解的分词方法和基于统计的分词方法。

1　常用中文分词算法

1.1　基于字符串匹配的分词算法

这种方法又叫做机械分词方法，它是按照一定的策略将待分析的汉字串与一个“充分大的”机器词典中的词条进行配，若在词典中找到某个字符串，则匹配成功。按照扫描方向的不同，串匹配分词方法可以分为正向匹配和逆向匹配；按照不同长度优先匹配的情况，可以分为最大匹配和最小匹配；按照是否与词性标注过程相结合，又可以分为单纯分词方法和分词与标注相结合的一体化方法。

1.2　基于理解的分词算法

这种分词方法是通过让计算机模拟人对句子的理解，达到识别词的效果。其基本思想就是在分词的同时进行句法、语义分析，利用句法信息和语义信息来处理歧义 现象。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，即它模拟了人对句子的理解过程。这种分词方法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，因此目前基于理解的分词系统还处在试验阶段。

1.3 基于统计的分词方法

从形式上看，词是稳定的字的组合，因此在上下文中，相邻的字同时出现的次数越多，就越有可能构成一个词。可以对语料中相邻共现的各个字的组合的频度进行统计，计算它们的互现信息。定义两个字的互现信息，计算两个汉字 X、Y 的相邻共现概率。互现信息体现了汉字之间结合关系的紧密程度。当紧密程度高于某一个阈值时，便可认为此字组可能构成了一个词。这里的假设是，用词造句无非是随机选词连在一块儿，是一个简单的一元过程。但实际上这种假设存在问题，如“的”字的单独出现概率太高了，它几乎总会从“的确”中挣脱出来。

2 基于 DP 的分词算法

先统计大量真实语料中各个词出现的频率，然后把每种分词方案中各词的出现概率乘起来作为这种方案的得分，不难求出得分最高的方案。以“其中最准确的就是动态规划的中文分词”这句话为例，动态规划的中文分词算法大致步骤如下：

1) 从最后一个字往第一个字处理。

2)最后一个字是“词”，查词频表，看它有没有出现，如果出现了，记下词频数，如果没出现，就认为它的词频数默认值是 1。然后计算概率，它的概率就是“词频数/157202”，157202 就是词频表里的词组总数。很显然，词频表没有这个字，于是它的概率就是 1/157202，记作 p[16] = 1/157202，p 表示概率，16 是这个字在句子里的坐标，这个句子一共 17 个字，坐标从 0 开始计算，所以它是第 16 位。

3) 倒数第二个字是“分”。这一次要处理的是两个字，也就是最后一个字和这一个字的组合，也就是“分词”，“分词”是全句的一部分，也就是子句。如何让“分词”这个子句的出现概率最高呢？先从“分”拆开，查它在词频表的词频，词频表没有，所以 p[15]=1/157202，如果把“分“和”词”拆开计算概率，那么子句的整体概率就是 p[15]*p[16]=(1/157202)*(1/157202)。如果从“词”拆，这么拆的话，“分词”就是一个整体了，查词频表，它的词频数是 390214，所以它的概率是 390214/157202，也就是说，子句的整体概率是 390214/157202，这个概率比前面的高，于是更新一下 p[15]=390214/157202。

....

以此类推，处理整个句子。

CSDN 上有 python 代码，详见附录[1].

3 DP 分词算法的改进可能

以上所有的分词算法都还有一个共同的大缺陷：它们虽然已经能很好地处理交集型歧义的问题，却完全无法解决另外一种被称为“组合型歧义”的问题。所谓组合型歧义，就是指同一个字串既可合又可分，究竟是合还是分，还得取决于它两侧的词语。到目前为止，所有算法对划分方案的评价标准都是基于每个词固有性质的，完全不考虑相邻词语之间的影响；因而一旦涉及到组合型歧义的问题，最大匹配、最少词数、概率最大等所有策略都不能实现具体情况具体分析。

于是我想到了一种基于动态规划的分词方法的改进可能，暂时未找到相关论文，因此不知道可行性如何。我们可以跳出一元假设。对于任意两个词语 w1 、 w2 ，统计在语料库中词语 w1 后面恰好是 w2 的概率 P(w1, w2) 。这样便会生成一个很大

的二维表。再定义一个句子的划分方案的得分为 P(∅, w1) • P(w1, w2) • … • P(wn-1, wn)，其中 w1, w2, …, wn 依次表示分出的词。甚至于对于总长度较小的句子的分词，我们可以类似的进行三元乃至更多元的统计，这样可以更加提高分词的准确度。

当然对于一些连人也无法准确理解的歧义句，如“乒乓球/拍卖/完了”和“乒乓球拍/卖完了”，我认为再完美的分词算法也无法解决。

对于一些特有名词，如地名和人名，可以通过在数据库中添加记录提高识别能力。

4 中文分词应用和展望

目前在自然语言处理技术中，中文处理技术比西文处理技术要落后很大一段距离，许多西文的处理方法中文不能直接采用，就是因为中文必需有分词这道工序。中文分词是其他中文信息处理的基础，搜索引擎只是中文分词的一个应用。其他的比如机器翻译（MT）、语音合成、自动分类、自动摘要、自动校对等等，都需要用到分词。因为中文需要分词，可能会影响一些研究，但同时也为一些企业带来机会，因为国外的计算机处理技术要想进入中国市场，首先也是要解决中文分词问题。在中文研究方面，相比外国人来说，中国人有十分明显的优势。

分词准确性对搜索引擎来说十分重要，但如果分词速度太慢，即使准确性再高，对于搜索引擎来说也是不可用的，因为搜索引擎需要处理数以亿计的网页，如果分词耗用的时间过长，会严重影响搜索引擎内容更新的速度。因此对于搜索引擎来说，分词的准确性和速度，二者都需要达到很高的要求。目前研究中文分词的大多是科研院校，清华、北大、中科院、北京语言学院、东北大学、IBM 研究院、微软中国研究院等都有自己的研究队伍。但科研院校研究的技术，大部分不能很快产品化，而一个专业公司的力量毕竟有限，看来中文分词技术要想更好的服务于更多的产品，还有很长一段路。

附录[1]

```
#分词函数
def solve(s):
    l = len(s)
    #p 表示概率，初始值是 0
    p = [0 for i in range(l+1)]
    #最后一位是为了方便计算，设置成 1
    p[l] = 1
    #词频数太大了，不能使用除法，会有精度问题，把分子分母分开存储
    div = [1 for i in range(l+1)]
    #记录拆分位置
    t = [1 for i in range(l)]
    #dp 算法做拆分
    for i in range(l-1,-1, -1):
        print "\ni = ", i
        for k in range(1, l-i+1):
            print " k = ", k
            if k > 1 and d[s[i:i+k］  == 1:
                print " continue"
                continue
            #判断子句的概率大小，以计算拆分位置
            if d[s[i:i+k］ *p[i+k]*div[i] > p[i]*d['_t_']*div[i+k]:
                p[i] = d[s[i:i+k］ *p[i+k]
                div[i] = d['_t_'] * div[i+k]
                t[i] = k
    i = 0
    while i < l:
        print s[i:i+t[i］ .encode("utf8"),
        i = i+t[i]

init()
s="   "#输入需要分词的句子
s=s.decode('utf8')
solve(s)
```

参考文献：

[1]曹卫峰. 中文分词关键技术研究[D].南京理工大学,2009.

[2]莫建文,郑阳,首照宇,张顺岚. 改进的基于词典的中文分词方法[J]. 计算机工程与设计，2013,(05):1802-1807.

[3] 周俊,郑中华,张炜. 基于改进最大匹配算法的中文分词粗分方法[J]. 计算机工程与应用,2014,(02):124-128.

[4] 龙树全,赵正文,唐华. 中文分词算法概述[J]. 电脑知识与技
术,2009,(10):2605-2607.

[5] 熊泉浩. 中文分词现状及未来发展[J]. 科技广场,2009,(11):222-225.