**综述报告题目：汉语分词研究**

**课程名称：大数据分析与应用**

**参与成员：秦博文、金瑞、朱远武**

**时间：2018年12月**

# 摘要

在我国，随着现代信息技术的发展，中文信息处理已经涵盖了字词、短语、句子、篇章等多层面的信息加工处理任务。汉语分词就成为中文信息处理的一项重要的基础性工作，许多中文信息处理项目均涉及到分词问题。本文首先对汉语分词技术进行了简要概述，包括介绍其定义、分词原因，分词存在的困难以及研究现状。重点介绍了机器学习和深度学习的汉语分词方法。

**关键词：**汉语、分词、机器学习、深度学习

# Abstract

In China, with the development of modern information technology, Chinese information processing has covered many levels of information processing tasks, such as words, phrases, sentences, texts and so on. Chinese word segmentation has become an important basic work of Chinese information processing. Many Chinese information processing projects involve word segmentation. Firstly, this paper gives a brief overview of Chinese word segmentation technology, including its definition, reasons, difficulties and research status. The Chinese word segmentation method for machine learning and deep learning is introduced.

**Key words: Chinese word segmentation、machine learning、deep learning**

**小组分工：**

|  |  |  |  |
| --- | --- | --- | --- |
| **姓名** | **学号** | **学院** | **分工** |
| **秦博文** | **3220180121** | **机电学院** | **一、二章** |
| **朱远武** | **3220180780** | **计算机学院** | **第四章** |
| **金瑞** | **3220180712** | **计算机学院** | **第三章** |

# 第1章 绪论

现代信息技术的发展要求用户有选择地阅读和处理海量文本,因此其预备工作——分词系统的研究是很有意义的。分词系统是中文信息处理中的一个主要组成部分,是中文自然语言理解、文献检索、搜索引擎以及文本挖掘系统中最基本的一部分[1]。为了进行中文信息小型化,需要提取关键知识,也就是说首先要分隔出单个的中文词语,然后才能进行词频统计、文本分类以及知识发现等工作,从而得到目标文献中用户需要的信息。

## 1.1 汉语分词定义

何谓汉语分词？汉语分词又名中文分词，指的是将一个汉字序列切分成一个个单独的词。分词就是将连续的字序列按照一定的规范重新组合成词序列的过程。我们知道，在英文的行文中，单词之间是以空格作为自然分界符的，而中文只是字、句和段能通过明显的分界符来简单划界，唯独词没有一个形式上的分界符，虽然英文也同样存在短语的划分问题，不过在词这一层上，中文比之英文要复杂得多、困难得多。例如：英文句子I am a teacher,如果用中文翻译过来就是“我是一名老师”。计算机可以简单通过空格知道teacher是一个单词，但是不能很容易明白”老“、”师“两个字结合起来才表示一个词。把中文的汉字序列切分成有意义的词，就是所谓的汉语分词。我是一名老师，分词的结果就是：我/是/一名/学生。

## 1.2 汉语分词的必要性

汉语的中文信息处理就是要“用计算机对汉语的音、形、义进行处理。”我们还知道,词是最小的能够独立活动的有意义的语言成分[2]。中文信息处理是指自然语言处理的分支，是指用计算机对中文进行处理。而计算机的所有语言知识都来自机器词典(给出词的各项信息)、句法规则(以词类的各种组合方式来描述词的聚合现象)以及有关词和句子的语义、语境、语用知识库。

然而,和大部分西方语言不同，书面汉语的词语之间没有明显的空格标记，句子是以字串的形式出现。因此对中文进行处理的第一步就是进行自动分词，即将字串转变成词串。切词体现了汉语与英语的显著的不同。英语文本是小字符集上的已充分分隔开的词串,而汉语文本是大字符集上的连续字串。可以这样设想汉语自动分词过程的困难如果把某个英语文本中的所有空格符都去掉,然后让计算机自动恢复文本中原有的空格符。实际上,这就是汉语词语的识别过程。

分词是汉语自然语言处理的第一步。目前,汉语自然语言处理的应用系统处理对象越来越多的是大规模语料,因此分词的速度和分词算法的易实现性变得相当关键。汉字的简体/繁 体转换、信息检索和信息摘录、搜索引擎、Web 文本挖掘、文本分类、文本校对等中文信息处理系统同样都首先需要分词作为其最基本的模块。

## 1.3研究背景、目的及意义

当代科技革命的主要特征，是以计算机为支持手段进行信息处理。在我国，随着信息时代的到来，可供人们查阅和检索的中文信息越来越多，各类信息混杂在一起，要想充分利用这些信息资源就要对它们进行整理，如果由人来做这项工作，已经是不可能的，而如果面对中文信息不采用分词技术，那么整理的结果就过于粗糙，而导致资源的不能充分利用，例如：“南京市长江大桥”和“重庆市长唐良智”中都有“市长”，而被当作同一类来处理，结果是检索“市长”的相关信息，会将他们都检索到，在信息量较少的情况下，似乎还能够忍受，如果是海量信息，这样的结果就会令人讨厌了。通过引入分词技术，就可以使机器对海量信息的整理更准确更合理。

中文信息处理技术是重要的计算机应用技术，它已经渗透到计算机应用的各个领域。如数据库技术、软件工程以及计算机网络等。故而在信息化时代的今天，中文信息的处理已经成为了我国信息化建设的“瓶颈“，国务院制定的国家中长期科技发展纲领中明确指出：“中文信息处理技术是高新技术发展的重点”，可见中文信息的处理已经得到国家的高度重视。因此，解决好中文信息的处理技术成为我国信息化进程中的“必决之役，必胜之战”。

而要正确处理好中文信息，首先得完成中文信息自动分词这项重要的基础性工作。中文分词，顾名思义，就是借助计算机自动给中文断句，使其能够正确表达所要表达的意思。中文不同于西文，没有空格这个分隔符，同时在中文中充满了大量的同义词，相近词，如何给中文断句是个非常复杂的问题，即使是手工操作也会出现问题。中文分词是信息提取、信息检索、机器翻译、文本分类、自动文摘、语音识别、文本语音转换、自然语言理解等中文信息处理领域的基础研究课题[3]。对于中文分词的研究对于这些方面的发展有着至关重要的作用。可以这样说，只要是与中文理解相关的领域，都是需要用到中文分词技术的。因此对于中文分词技术的研究，对于我国计算机的发展有着至关重要的作用。

## 1.4 汉语自动分词的研究现状

最早的中文分词方法是由北京航空航天大学的梁南元教授提出的一种基于“查字典”分词方法。该方法的思想事把整个中文句子，读一遍，然后把字典里有的词都单独标示出来，当遇到复合词的时候，就找到最长的词匹配，遇到不认识的字符串就分割成单个文字。这种分词方法效率并不高，但它的提出为中文分词技术奠定了基础[4]。

中文自动分词己经被研究了二十多年，但是目前仍然是制约汉语信息处理发展的一个瓶颈。它主要存在语言学和计算机科学等两方面的困难。

语言学方面的困难有：

（1）汉语分词还没有形成一个公认的分词标准，这也是计算机和人共同面临的一个问题，同一个文本可能被不同的人划分为几种不同的分词结果。

（2）词的定义不统一，语言学界对词还没有给出一个为大家广泛接受的、严格且统一的非形式定义。

（3）汉语词与词之间不像西文那样有明显的分隔符（如空格等），也不像日文那样有丰富的词尾变化（如日语中的片假名和平假名构成了书面日语的词尾变化）。

（4）词的具体判定问题还没有完全解决。尽管《 信息处理用现代汉语分词规范》提出了分词单位和一套比较系统的分词规则，但是由于真实文本的复杂性和多样性，实践与理论之间的重大差异，仍然没有能够有彻底的解决办法。

（5）新词在不断增加（如人名、地名、新产生的词、外来词等。

计算机方面的困难：

（1）如何对语义进行理解和形式化。

（2）没有合理的自然语言形式模型。

（3）如何有效地利用和表示分词所需的语法知识和语义知识。

此外，自80年代初中文信息处理领域提出了自动分词以来，研究者在对汉语分词技术的研究过程中，开发出一些实用性的分词系统，其中几个比较有代表性的自动分词系统在当时产生了较大的影响。

（1）CDWS分词系统

CDWS分词系统是我国第一个实用的自动分词系统，由北京航空航天大学计算机系于1983年设计实现，它采用的自动分词方法为最大匹配法，辅助以词尾字构词纠错技术。其分词速度为5-10字/秒，切分精度约为1/625。

（2）SCWS分词系统

Hightman开发的一套基于词频词典的机械中文分词引擎，它能将一整段的汉字基本正确的切分成词。采用的是采集的词频词典，并辅以一定的专有名称，人民，地名，数字年代等规则识别来达到基本分词，经小范围测试大概准确率在90%到95%之间，已基本满足一些小型搜索引擎、关键字提取等场合运用。45Kb左右的文本切词时间是0.026秒，大概是1.5MB文本/秒，支持PHP4和PHP5。

（3）ICTCLAS分词系统

这是最早的中文开源分词项目之一，ICTCLAS在国内973专家组组织的评测中活动获得了第一名，在第一届国际中文处理研究机构SigHan组织的评测中都获得了多项第一名。ICTCLAS3.0分词速度单机996KB/s，分词精度98.45%，API不超过200KB，各种词典数据压缩后不到3M.ICTCLAS全部采用C/C++编写，支持Linux、FreeBSD及Windows系列操作系统，支持C/C++、C#、Delphi、Java等主流的开发语言。

（4）ABWS分词系统

ABWS分词系统是山西大学计算机系研制的自动分词系统，系统使用“两次扫描联想-回溯”方法，运用了较多的词法、句法等知识。其切分正确率为98.6%，运行速度为48词/分钟。

（5）HTTPCWS分词系统

HTTPCWS 是一款基于HTTP协议的开源中文分词系统，目前仅支持Linux系统。HTTPCWS 使用“ICTCLAS 3.0 2009共享版中文分词算法”的API进行分词处理，得出分词结果。HTTPCWS 将取代之前的 PHPCWS 中文分词扩展。

（6）CASS分词系统

CASS分词系统是北京航空航天大学余1988年实现的分词系统。它使用正向增字最大匹配，运用知识库来处理歧义字段。其机械分词速度为200字/秒以上，知识库分词速度150字/秒。

（7）CC-CEDICT分词系统

一个中文词典开源项目，主要功能：提供一份以汉语拼音为中文辅助的汉英辞典，截至2009年 2 月 8 日，已收录 82712个单词。其词典可以用于中文分词使用，而且不存在版权问题。Chrome中文版就是使用的这个词典进行中文分词的。

（8）Paoding

Paoding（庖丁解牛分词）是一个使用Java开发的，可结合到Lucene应用中的，为互联网、企业内部网使用的中文搜索引擎分词组件。Paoding填补了国内中文分词方面开源组件的空白，致力于此并希翼成为互联网网站首选的中文分词开源组件。Paoding中文分词追求分词的高效率和用户良好体验。

Paoding's Knives中文分词具有极高效率和高扩展性。引入隐喻，采用完全的面向对象设计，构思先进。

高效率：在PIII 1G内存个人机器上，1秒可准确分词100万汉字。

采用基于不限制个数的词典文件对文章进行有效切分，使能够将对词汇分类定义。能够对未知的词汇进行合理解析。

（9）NLTK——斯坦福中文分词器

斯坦福大学自然语言处理组是世界知名的NLP研究小组，他们提供了一系列开源的Java文本分析工具，包括分词器(Word Segmenter)，词性标注工具（Part-Of-Speech Tagger），命名实体识别工具（Named Entity Recognizer），句法分析器（Parser）等，可喜的是，他们还为这些工具训练了相应的中文模型，支持中文文本处理。

（10）盘古分词系统

盘古分词是一个基于.net平台的开源中文分词组件，提供lucene(.net 版本) 和HubbleDotNet的接口。

高效：Core Duo 1.8 GHz 下单线程分词速度为390K字符每秒

准确：盘古分词采用字典和统计结合的分词算法，分词准确率较高。

功能：盘古分词提供中文人名识别，简繁混合分词，多元分词，英文词根化，强制一元分词，词频优先分词，停用词过滤，英文专名提取等一系列功能。

（11）MMSEG4J

MMSEG4J基于Java的开源中文分词组件，提供lucene和solr接口。一方面，MMSEG4J是用Chih-Hao Tsai的MMSEG算法实现的中文分词器，并实现lucene的analyzer和solr的TokenizerFactory以方便在Lucene和Solr中使用。另一方面，MMSEG算法有两种分词方法：Simple和Complex,都是基于正向最大匹配。Complex加了四个规则过滤。官方说：词语的正确识别率达到了98.41%。MMSEG4J已经实现了这两种分词算法。

（12）复旦分词系统

此系统由四个模块构成。一、预处理模块，利用特殊的标记将输入的文本分割成较短的汉字串，这些标记包括标点符号、数字、字母等非汉字符，还包括文本中常见的一些字体、字号等排版信息。二、歧义识别模块，使用正向最小匹配和逆向最大匹配对文本进行双向扫描，如果两种扫描结果相同，则认为切分正确，否则就判别其为歧义字段，需要进行歧义处理；三、歧义字段处理模块，此模块使用构词规则和词频统计信息来进行排除歧义。最后，此系统还包括一个未登录词识别模块，实验过程中，对中文姓氏的自动识别达到了70%的准确率。系统对文本中的地名和领域专有词汇也进行了一定的识别。

## 1.5 本文研究内容

本文主要对汉语分词技术进行了仔细调研，及时准确地掌握了分词系统的现状，分析了分词技术目前存在的难点以及分词算法的工作原理。对汉语分词技术作了一次较为系统的总结，本文的章节安排如下：

首先从目前语言信息处理的瓶颈入手，提出了所要研究的自然语言的处理的关键技术——汉语分词。接下来通过第一章介绍了汉语分词基本概念，汉语分词目前的研究背景、意义及目的；第二章对汉语分词进行了介绍，包括汉语分词的问题描述，汉语分词的难点，以及汉语分词现有的算法思想和优缺点等内容。第三章介绍了机器学习的分词方法；第四章主要介绍了基于深度学习的汉语分词方法。

# 第2章 汉语分词简介

汉语以字为基本书写单位，词语之间没有明显的区分标记。而汉语分词技术作为中文信息处理的关键性基础技术，通俗地讲就是由计算机在中文文本中的词与词之间加上标记。汉语分词的研究已经持续了三十多年，可是到现在仍然没有形成一个很好的中文分词系统，若要推进我国信息化的进程就不得不继续对汉语分词系统进行更加深入的研究，本章主要对汉语分词进行了介绍。

## 2.1 汉语分词问题描述

在信息检索、语音识别、机器翻译等技术领域中通常要对中文的每一句话进行理解，也就是要对每一个词进行识别和理解，从而完成相应的指令。那么，就需要将句子中的词单独切分出来进行识别，这就是汉语分词技术。由于汉语对于词的界限较为模糊，如何分词，什么样的叫词，都需要一个专门的词库来进行区分，不过到目前为止并没有这么一个完善的词库来帮助我们界定词的界限[5]。

所以目前汉语分词存在两大基本问题，也是汉语分词的难点，一个是歧义识别问题，另一个则是未登录词（新词）问题。

歧义识别问题是由于中文自身特点所决定的，对于中文的一句话不同的切分可能得到不同的意思，例如，“南京市长江大桥”，可以切分成“南京市/长江/大桥”，也可以切分成“南京/市长/江大桥”。

对于未登录词，则是由于语言在不断的发展和变化所导致的。也就是那些在分词词典中没有收录，但又确实能称为词的那些词。最典型的是人名，人可以很容易理解，但是对于计算机来说识别起来是有一定难度的。

## 2.2 汉语分词的难点分析

汉语分词研究了近30年，虽然取得了一些成就，并且也有了成熟的分词算法，。但是中文是一种十分复杂的语言，而中文分词的基础性问题，也是关键性问题并没有解决。也就是歧义识别问题和未登录词的识别问题，依然成文汉语分词技术发展的两大难点。

### 2.2.1 歧义识别

歧义是指同样的一句话，可能有两种或者更多的切分方法。典型的歧义有交集型歧义（约占全部歧义的85%以上）、组合型歧义以及混合型歧义[6]。交集型歧义是这样一种歧义：汉字串AJB被称作交集型切分歧义，如果满足AJ、JB同时为词（A、J、B分别为汉字串）。例如：表面积，因为“表面”和“面积”都是词，那么这个短语可以切分成“表面/积”和“表/面积“。组合型歧义则是这样一种歧义：汉字串AB被称作多义组合型切分歧义，如果满足A、B、AB同时为词。例如：“这个人/手上有痣”和“公司缺人手“。所谓混合型歧义则是同时包含交集型歧义和组合型歧义。例如：”这样的人才能经受住考验“这句话可以切分成”这样的/人/才能/经受住考验“、”这样的/人才/能/经受住考验“以及”这样的/人/才/能/经受住考验。

如果交叉歧义和组合歧义计算机都能解决的话，在歧义中还有一个难题，就是真歧义。真歧义意思是给出一句话，由人去判断也不知道哪个应该是词，哪个应该不是词。例如：“乒乓球拍卖完了”，可以切分成“乒乓/球拍/卖/完/了”、也可切分成“乒乓球/拍卖/完/了”，如果没有上下文其他的句子，恐怕谁也不知道“拍卖”在这里算不算一个词。

### 2.2.2 未登录词识别

命名实体（人名、地名）、新词，专业术语称为未登录词。也就是那些在分词词典中没有收录，但又确实能称为词的那些词。最典型的是人名，人可以很容易理解。句子“王军虎去广州了”中，“王军虎”是个词，因为是一个人的名字，但要是让计算机去识别就困难了。如果把“王军虎”做为一个词收录到字典中去，全世界有那么多名字，而且每时每刻都有新增的人名，收录这些人名本身就是一项既不划算又巨大的工程。即使这项工作可以完成，还是会存在问题，例如：在句子“王军虎头虎脑的”中，“王军虎”还能不能算词？

除了人名以外，还有机构名、地名、产品名、商标名、简称、省略语等都是很难处理的问题，而且这些又正好是人们经常使用的词，因此对于搜索引擎来说，分词系统中的新词识别十分重要。新词识别准确率已经成为评价一个分词系统好坏的重要标志之一。

### 2.2.3 现有的研究成果

#### 2.2.3.1 歧义消除

解决歧义的方法可以分为两类：基于规则的方法和基于统计的方法。

（1）基于规则的方法：

李家福提出了基于规则的消除切分歧义的方法，并根据句法、语义规则和语法、语义解析进行分词判断。这些规则仅涉及若干毗邻词之间的线性关系，没有反应句字中各成分之间的层次关系，可靠性不强，难以建立完整、有效、无矛盾的体系。

（2）基于统计的方法：

谭琼提出了采用双向扫描法识别歧义字段，然后利用互信息和T-测试解决歧义。但是需要进一步解决下面的问题：①对r存在多处切分歧义的交集型字段，如何计算每种切分的可能性；②当每个句子都有多种切分方法时，该方法的计算量较大。

孙茂松提出了利用句内相邻字之间的互信息和T测试这两个统计量来解决汉语自动分词中交集型歧义切分字段的方法。其优点是：①不需要人工标注语料，直接从生语料库出发，通过字的统计信息模拟词频，进而设计交集型歧义切分字段的算法；②字的统计信息获取过程是全自动的，因而避免了人工标注语料可能引起的各种问题，保证了数据的准确性、一致性、方法的简明性和移植性。存在的不足是：①连续字对的出现与语料的大小不一定有直接的关系；②必须保证训练语料要足够的大；③二元语法模型的固有缺陷导致了那些不常见用法的错误。例如：化工厂分为：化工/厂而不是：化/工厂。

#### 2.2.3.2 未登录词的处理方法

产生未登录词的原因主要有：①机器可读词典中词目的选择和词目的数量；②机器可读词典与待处理文本中的词汇的匹配关系，包括机器可读词典对待处理文本中词汇的覆盖率。覆盖率指待处理文本的词汇在机器可读词典中所占的比例。如果待处理文本中含有m个不重复出现的词，其中有n个词在机器可读词典中出现，则机器可读词典对待处理文本词汇的覆盖率为n/m。

目前主要有基于分解与动态规划策略的汉语未登录词识别和基于语料学习的未登录词检测等方法。

Keh-Jiann Chen提出了一种基于语料学习的未登录词检测方法。其方法能够提取那些用来区分单音节词和多音节词词素的语法规则。该方法的优点是：①进行自动规则学习；②自动评测每个规则的性能；③通过动态选择规则集，能够平衡召回率和正确率。但是，这些优点是依赖于语料库的。该文也没有给出如何提取规则和如何运用规则来检测未登录词。同时，也需要考虑由多音节构成的未登录词的识别方法。

## 2.3 主要分词算法

从开始研究中文分词算法到现在，虽然没有出现非常完美的分词算法，但是也还是出现了许多比较好的分词算法，目前的分词算法主要包含基于字典的分词算法，基于统计的分词算法和基于理解的分词算法，下面简要介绍一下这些算法。

### 2.3.1 基于字典的分词方法

基于字典的分词方法又叫机械分词法是在20世纪50年代末由苏联专家提出的最早出现的一种自动分词算法。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（识别出一个词）[7]。

按照扫描方向的不同，字符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（最长）匹配和最小（最短）匹配；按照是否与词性标注过程相结合，可以分为单纯分词方法和分词与词性标注相结合的一体化方法。

**优缺点**：机械匹配算法简洁、易于实现．其中的代表算法——最大匹配法，体现了长词优先的原则，在实际工程中应用最为广泛。但其局限也是很明显的：效率和准确性受到词库容量的约束；机械匹配算法采用简单机械的分词策略，不涉及语法和语义知识，所以对于歧义切分无法有效地克服。切分精度不高。

### 2.3.2 基于统计的分词算法

基于统计的分词方法是在给定大量已经分词的文本的前提下，利用统计机器学习模型学习词语切分的规律（称为训练），从而实现对未知文本的切分。例如最大概率分词方法和最大熵分词方法等。随着大规模语料库的建立，统计机器学习方法的研究和发展，基于统计的中文分词方法渐渐成为了主流方法。

主要的统计模型有：N元文法模型（N-gram），隐马尔可夫模型（Hidden Markov Model，HMM），最大熵模型（ME），条件随机场模型（Conditional Random Fields，CRF）等。

在实际的应用中，基于统计的分词系统都需要使用分词词典来进行字符串匹配分词，同时使用统计方法识别一些新词，即将字符串频率统计和字符串匹配结合起来，既发挥匹配分词切分速度快、效率高的特点，又利用了无词典分词结合上下文识别生词、自动消除歧义的优点。

### 2.3.3 基于理解的分词算法

基于理解的分词方法又称基于人工智能的分词方法，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。

# 第三章 汉语分词算法

## 3.1 基于机械匹配的分词算法

机械匹配法是自动分词中最基础的算法，其基本思想是：

事先建立一个词库，其中包含所有可能出现的词

对给定的待分词汉字串，按照某种确定的原则切取子串，若该子串与词库中的某词条相匹配，则该子串是词，继续分割剩余的部分直到剩余部分为空

否则，该子串不是词，转到（2）重新切取*S*的子串进行匹配

其中，根据切取子串的不同原则，可以将机械匹配法分为几类，一种是按切取子串的方向来分，可以分为正向匹配法和逆向匹配法；第二种是按每次匹配时优先考虑长词还是短词来分，可分为最大匹配法和最小匹配法；第三种是按匹配不成功时重新切取的策略来分，可分为增字法和减字法。最后有一种N-最短路径法，是对机械匹配法的优化。

机械匹配算法简洁而且易于实现，在实际工程应用最为广泛。其实现比较简单，但局限也很明显，效率和准确性受到词库容量的约束；因为采用的是简单机械的分词策略，不涉及语法和语义知识，所以对于歧义切分无法有效的克服，切分精度不高。现阶段有不少改善机械匹配性能方法，但从整体上来看，单纯的采用机械匹配式的方法是无法满足中文信息处理中对于汉语分词的要求的，仍然需要在机械匹配分词的基础上利用语言信息来进行歧义校正。

因此，机械匹配算法的优缺点其实很明显。优点：首先，可以确保使用基于词典的分词方法所切分出来的中文字符串百分百是“词语”，因为切分出来的字符串全是和词典匹配得到的；其次，进行中文分词时只需要一个中文词典而不需要额外的语料集，不需要建立额外的语言模型，没有非常复杂的计算，计算量相对较少，因此分词的效率相对较高。缺点：因为基于词典的分词方法只是单纯地在词典中进行中文字符串的匹配工作，某个字符串在词典中匹配到了，便认为它是一个词语，匹配不到，便认为它不是一个词语，根本没有考虑词语与词语之间的关系，也没有进行语法方面的考量，所以容易出现歧义。

接下来介绍几种常用机械匹配算法：

### 3.1.1 正向最大匹配法

正向最大匹配法（Forward maximum Matching Method），也称为FMM方法。其算法描述如下：

假设句子: , 某一词：, 为词典中最长词的字数。

1. 令,当前指针指向输入字串的初始位置,执行下面的操作:
2. 计算当前指针到字串末端的字 数(即未被切分字串的长度),如果 ,转(4),结束算法。否则,令词典中最长单词的字数,如果,令;
3. 从当前起取个汉字作为词,判断:
4. 如果确实是词典中的词,则在后添加一个切分标志,转 (c);
5. 如果不是词典中的词且的长度大于1,将从右端去掉一个字,转(a)步;否则(的长度等于1),则在后添加一个切分标志,将作为单字词添加到词典中,执行(c)步;
6. 根据wi的长度修改指针 的位置,如果指向字串末端,转 （4）,否则, ,返回 （2）;
7. 输出切分结果,结束分词程序。

其流程图如图1：



图1 FMM流程图

例：他是研究生物化学的。

切分过程：1. 他/是研究生物化学的。

2. 他/是/研究生物化学的。

3. 他/是/研究生/物化学的。

4. 他/是/研究生/物/化学的。

5. 他/是/研究生/物/化学/的。

### 3.1.2 逆向最大匹配法

逆向最大匹配法（The Reverse Directional Maximum Matching Method），也称为RMM方法。它与FMM最大的不同在于，切分汉字时，RMM不是按照汉字顺序从左到右依次抽取子串，而是从汉字尾端开始抽取。

RMM的流程图如图2：



图2 RMM流程图

RMM与FMM的原理基本相同，其本质只是一个方向上的区别。然而一般来说，RMM的准确率却略过高于FMM，由于大家习惯都是正向的来理解句子，所以RMM的错误率会稍小。据网上的统计数据表明，单纯使用FMM的错误率是1/169，单纯使用RMM的错误率是1/245。

例：他是研究生物化学的。

切分过程：1. 他是研究生物化学/的。

2. 他是研究生物/化学/的。

3. 他是研究/生物/化学/的。

4. 他是/研究/生物/化学的。

5. 他/是/研究/生物/化学/的。

### 3.1.3双向扫描法

双向扫描法是将FMM得到的分词结果和RMM得到的结果进行对比，从而决定正确的分词方法。据研究表明，中文90.0%左右的句子，FMM和RMM完全重合且正确，还有9.0%的句子得到的结果不一样，但其中必定有一个是正确的；还有剩下1.0%左右的句子是两者切分结果重合却错误或者两个切分不一致都错误的。

因为有了如此高的正确率，双向扫描法在实际应用中广泛的得到使用，如果两种切分方法不一致，就采用其他方式去在进行切分，如人工干预、频度标记的算法、结合上下文相关信息的算法等。

不过虽然提高了分词的准确率，降低了一定的错误率，但也同时付出了一定的代价：一个时间上的：同时执行两种算法，增加了总体执行时间，导致算法速度相对较低；另一个是空间上的：两种算法执行时需要的可能是不同的词典结构，这就需要加载两个不同的词典，从而加大了内存空间的使用。

双向扫描法的流程图如图3：



图3 双向匹配法流程图

例：泰国人民很友好。

切分结果FMM：泰国人/民/很/友好。×

切分结果RMM：泰国/人民/很/友好。√

### 3.1.4 N-最短路径法

N-最短路径法是对最短路径法的一种改进，进本思想是根据现有词典，将每个句子分解为一个带权有向无环图，每个字代表图的点，边代表可能的分词，边的起点为词的第一个字，终点为词尾的一下个字，这里一般用词的频率来表示边的权值，最终的结果即在上述带权有向无环图中寻找N条权值和最大的路径。

我们可以建立如下模型，设字串,其中为单个的字，字串的长度为，。建立有向无环图G，节点数位，节点编号依次为。

通过以下两个步骤建立G所有可能的词边，

1. 相邻节点之间建立有向边 <>,边对应的词默认为。
2. 如果 是一个词,则节点之间建立有向边 <>,边对应的词为。

假定词与词之间是相互独立的，这里引入词的出现概率，得到一个基于N-最短路径算法的一元统计模型。根据大数定理，在样本数据很大时，样本的频率接近其概率值，所以，的极大似然估计值等于词频，则，其中为在训练样本中出现的次数。最终可以得到边长公式为：

由公式可知的词频越高，边长越短。如此，字串和它所包含的词与有向无环图G的边一一对应，最后用Dijkstra算法计算最短路径，便可以得到最终结果。

以“今天下午休息。”为例，为方便计算，假设所有边的长度均为1。

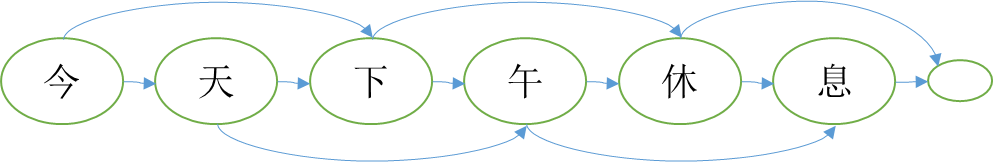


图4 最短路径示例

如图4，每个相邻的字之间都有一条路径，“今天”，“天下，“下午”，“午休”，“休息”都可以组成词语并假设他们都是字典中存在的词，则每个词都增加一条路径，箭头指向词尾的下一个字，再根据Dijkstra计算路径长度。

可以得到如下的结果：

表1 分词结果和长度示例

|  |  |  |
| --- | --- | --- |
| 序号 | 分词结果 | 路径长度 |
| 1 | 今天/下/午/休息/ | 4 |
| 2 | 今/天/下午/休息/ | 4 |
| 3 | 今/天/下/午/休息/ | 5 |
| 4 | 今/天/下/午休/息/ | 5 |
| 5 | 今/天/下午/休/息/ | 5 |
| 6 | 今/天下/午休/息/ | 4 |
| 7 | 今天/下午/休息/ | 3 |

根据分词结果，取前N种路径最短的分词作为粗分结果，再用其他方式进行精分。

## 3.2 基于统计语言模型的汉语分词算法

基于统计的分词方法摒弃了词典，它在进行分词的时候不许需要词典作为输入，而是需要各种各样的语言模型作为输入，而语言模型的训练需要的中文语料集。所谓中文语料集是指的大规模语料集，是包含大规模（一般最低在十万数量级以上）的中文句子的文档。

之所以需要大规模语料集，其主要思想是：中文句子在上下文中，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。

基于统计的中文分词的算法其基本步骤如下：

* 1. 加载大规模中文语料集
  2. 训练语言模型
  3. 进行中文分词

### 3.2.1 N元语言模型

对于一个中文字符串，可以将其看成一个连续的字符串序列，他可以是单字符序列，也可以是词序列，对于字符串S有一种切分方式为：，N元语言模型计算的就是字符串被切分为：的概率有多大，此时将其出现的概率记为，即。

对于句子中词语每个词语，N元语言模型假定出现的概率与前面个词语有关，并且概率值使用条件概率公式计算，于是有下面的公式：

更进一步的，可以将上述公式演变为下面的形式：

在上述表达式中，对于语句产生的概率是由产生第一个字的概率，乘以在第一个字产生的条件下产生第二个字的概率，再给定前两个字的条件下产生第三个字的，一次类推，知道乘以在给定前个单字的条件下产生的概率。由此可见，每一个词语，其出现的概率是由该词语前面的个词语所决定的。这是N元语言模型的核心思想，即用已知的条件来推测未知的。但在实际的应用中，这样做会有两个弊端：一个是要进行计算的句子可能会很长，包含的词语数目很多，这时，如果假设第个词语的出现和前个词语都有关系的话，在计算未知靠后的词语的概率的时候，计算量就会很大；二是在一般的情况下，第个词的出现只是和其相邻的几个词语的关联比较大，如果钱个都算进去的话，会导致有很多干扰数据的产生，计算结果会偏离实际情况。因此在实际使用中会根据资源的限制等条件来选择参数N，对应N的不同有不同语言模型。这就是N阶的语言模型。

例如之前给的一个句子：他是研究生物的。

有如下两种分词方式，哪种方式是正确的呢？

（1）他|是|研究生|物|的

（2）他|是|研究|生物|的

若我们采用二元文法，则需要计算下面两个式子，对比结果

1：𝑝(𝑆𝑒𝑔1)=𝑝(他│<𝐵𝑂𝑆>)×𝑝(是│他)×𝑝(研究生│是)×𝑝(物│研究生)×𝑝(的│物)×𝑝(的│<𝐸𝑂𝑆>)

2：𝑝(𝑆𝑒𝑔2)= 𝑝(他│<𝐵𝑂𝑆>)×𝑝(是│他)×𝑝(研究│是)×𝑝(生物│研究)×𝑝(的│生物)×𝑝(的│<𝐸𝑂𝑆>)

### 3.2.2 互信息模型

N元语言模型中用到了大规模语料库，那么有一个必须要解决的问题就是如何用语料库训练出来一个词典，找出在语料集中的“词语”。互信息解决的就是这样一个问题。

互信息模型的核心思想是：根据语料库中字与字之间的“紧密”程度来判断其组合是否为词语。如果两个字总是紧挨着出现，那么这两个字在很大程度上就可以判定为是一个中文词语，如果两个字没有相邻出现过或者只是偶尔相邻出现了一次或者几次，而单独出现了很多次，那么便可以得出这两个字之间的“紧密度”并不是那么强的结论，从而可以判定这两个字不能组成一个词语。

那么具体如何去判断两个字的紧密程度呢？我们通过以下几个公式推导：

信息论中的自信息公式是：

概率本身是对事件的确定性度量，而信息是对事件的不确定性度量，因此，做某件事情的概率越低，信息量越大。当两个事件相对独立时，他们的联合事件的信息量就等于各自信息量的和。用公式表示为

在信息论中互信息公式是：

与分词系统相结合时，互信息是用来表示两个字之间的结合的强度，其公式为：

其中的就是代表被研究的对象，代表的是同时出现的概率，是汉字出现的概率。是汉字出现的概率，结社他们在语料库中出现的次数为，语料库中的词频总数位，这里采用样本频率代替概率即：，，。

互信息描述的是两个字之间的紧密程度，，根据互信息的公式即

进而推导出 ，与之间是正相关的，MI值越大表示两个字之间的结合越紧密。反之，断开的可能性就越大。当MI的值大于某个阈值的时候，就可以认为是基本构成词的；，则认为的关系很弱，基本是不相关的；若，就认为基本是没有关系的，也就是这两个字互斥，基本不会成词。

### 3.2.3 最大熵模型

熵是对一个随机事件的不确定性的衡量，其计算公式为：

最大熵模型的作用是在已知条件下选择一个适合的分布来预测可能出现的时间，最重要的思想是在只掌握关于未知分布的部分知识时，应选取符合这些知识但熵值最大的概率分布。在实验结束前是实验结果不确定性的度量，实验完成后是从实验中获取的信息量。越大代表不确定性越大，实验结束后从中得到的信息量也越大。

用浅显的说法解释熵的含义：不要把鸡蛋放在同一个篮子里。也就是说鸡蛋分布越平均，危险性就越少，最大熵的原理就是容纳所有的不确定性，把这件事件的风险降到最低。即在知道有限知识的情况下，预测未知的事件是不做偏移的假设。

从概率论的角度说就是，将已知事件作为约束条件，求得可以使熵最大化的概率分布。

在最大熵模型中，目标是构造一个能生成训练样本分布的统计模型，建立特征方程。该特征必须能较完整地表达训练样本中数据的特性。

我们以英汉翻译为例：对于英语中的“take”，它对应汉语的翻译有：

(t1)“抓住”：The mother takes her child by the hand.母亲抓住孩子的手。

(t2)“拿走”：Take the book home. 把书拿回家。

(t3)“乘坐”：to take a bus to work. 乘坐公共汽车上班。

(t4)“量”：Take your temperature. 量一量你的体温。

(t5)“装”：The suitcase wouldn’t take another thing. 这个衣箱不能装别的东西了。

(t6)“花费”：It takes a lot of money to buy a house. 买一所房子要花一大笔钱。

(t7)“理解、领会”：How do you take this package? 你怎么理解这段话？

假设对于所有的英文”take”，只有这七种翻译。则存在着如下限制： ，(1)

表示在一个含有单词take的英文句子中，take翻译成 的概率。

在这个限制下，对每种翻译赋予均等一致的几率为：

但是对于“take”，我们通过统计发现它的前两种翻译(t1)和(t2)是常见的，假设满足如下条件

，(2)

在(1)和(2)共同限制下，分配给每个翻译的概率分布形式有很多。但是最一致的分布为：

可以验证，最一致的分布具有最大的熵值。

但是上面的限制，都没有考虑上下文的环境，翻译效果不好。因此我们引入特征。例如，英文“take”翻译为 “乘坐”的概率很小，但是当“take”后面跟一个交通工具的名词“bus”时，它翻译成“乘坐”的概率就变得非常大。为了表示 take 跟有“bus”时翻译成“乘坐”的事件，我们引入二值函数：

，(3)

表示上下文环境，这里看以看作是含有单词 take 的一个英文短语，而代表输出，对应着“take”的中文翻 译。^next(x)看作是上下文环境的一个函数，表示中跟在单词 take 后的一个单词为“bus” 。这样一个函数我们称作一个特征函数，或者简称一个特征。引入诸如公式(3)中的特征，它们对概率分布模型加以限制，求在限制条件下具有最一致分布的模型，该模型熵值最大。

这就是最大熵模型。

### 3.2.3 隐马模型

隐马模型（Hidden Markov Model；HMM）是经典的描述随机过程的统计方法，在自然语言处理中得到了广泛的应用。

举一个经典的例子：一个东京的朋友每天根据天气{下雨，天晴}决定当天的活动{公园散步,购物,清理房间}中的一种，我每天只能在twitter上看到她发的推“啊，我前天公园散步、昨天购物、今天清理房间了！”，那么我可以根据她发的推特推断东京这三天的天气。在这个例子里，显状态是活动，隐状态是天气。

HMM其形式主要是一个五元组：

其中，Q是有限状态集合（隐状态），V是观测结果有限集（观测序列），A是状态转移矩阵（转移概率），B是状态到观测值的概率矩阵（发射概率），π是初始状态分布（初始概率）。

如何求解Q是在汉语分词中HMM的主要应用。即给定一个字的序列，找出最可能的标签序列（断句符号：[词尾]或[非词尾]构成的序列）。结巴分词目前就是利用BMES标签来分词的，B（开头）,M（中间),E(结尾),S(独立成词）

那么具体如何使用隐马模型进行分词呢？首先规定每个字有4个词位：词首 B，词中 M，词尾 E，单字成词 S

由于HMM是一个生成式模型，X为观测序列，Y为隐序列。

HMM有三类基本问题：

预测(filter)：已知模型参数和某一特定输出序列，求最后时刻各个隐含状态的概率分布，即求 。通常使用前向算法解决.

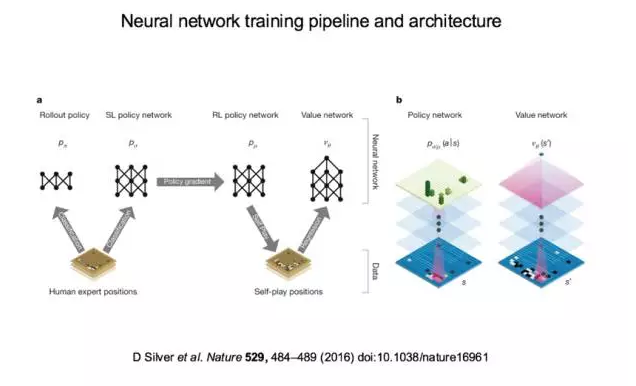
平滑(smoothing)：已知模型参数和某一特定输出序列，求中间时刻各个隐含状态的概率分布，即求 。通常使用forward-backward 算法解决.

解码(most likely explanation): 已知模型参数，寻找最可能的能产生某一特定输出序列的隐含状态的序列. 即求 , 通常使用Viterbi算法解决.

分词就对应着HMM的解码问题，模型参数(转移矩阵，发射矩阵)可以使用统计方法计算得到，原始文本为输出序列，词位是隐状态序列，因此，使用Viterbi算法求解即可。

# 第四章 深度学习在中文分词中的应用

随着AlphaGo的大显神威，Deep Learning（深度学习）的热度进一步提高。深度学习来源于传统的神经网络模型。传统的神经网络一般由输入层，隐藏层，输出层组成，其中隐藏层的数目按需确定。深度学习可以简单的理解为多层神经网络，但是深度学习的却不仅仅是神经网络。深度模型将每一层的输出作为下一层的输入特征，通过将底层的简单特征组合成为高层的更抽象的特征来进行学习。在训练过程中，通常采用贪婪算法，一层层的训练，比如在训练第k层时，固定训练好的前k-1层的参数进行训练，训练好第k层之后的以此类推进行一层层训练。



深度学习在很多领域都有所应用，在图像和语音识别领域中已经取得巨大的成功。从2012年开始，LSVRC（LargeScale Visual Recognition Challenge）比赛中，基于Deep Learningd计算框架一直处于领先。2015年LSVRC的比赛中，微软亚洲研究院（MSRA）在图像检测（Objectdetection），图像分类定位（Object Classification+localization）上夺冠，他们使用的神经网络深达152层。

## 4.1 深度学习在NLP中的应用

在自然语言处理上，深度学习在机器翻译、自动问答、文本分类、情感分析、信息抽取、序列标注、语法解析等领域都有广泛的应用。2013年末google发布的word2vec工具，可以看做是深度学习在NLP领域的一个重要应用，虽然word2vec只有三层神经网络，但是已经取得非常好的效果。通过word2vec，可以将一个词表示为词向量，将文字数字化，更好的让计算机理解。使word2vec模型，我们可以方便的找到同义词或联系紧密的词，或者意义相反的词等。

## 4.2 词向量介绍

词向量的意思就是通过一个数字组成的向量来表示一个词，这个向量的构成可以有很多种。最简单的方式就是所谓的one-hot向量。假设在一个语料集合中，一共有n个不同的词，则可以使用一个长度为n的向量，对于第i个词（i=0…n-1），向量index=i处值为1外，向量其他位置的值都为0，这样就可以唯一的通过一个[0，0，1，…，0，0]形式的向量表示一个词。one-hot向量比较简单也容易理解，但是有很多问题，比如当加入新词时，整个向量的长度会改变，并且存在维数过高难以计算的问题，以及向量的表示方法很难体现两个词之间的关系，因此一般情况下one-hot向量较少的使用。

如果考虑到词和词之间的联系，就要考虑词的共现问题。最简单的是使用基于文档的向量表示方法来给出词向量。基本思想也很简单，假设有n篇文档，如果某些词经常成对出现在多篇相同的文档中，我们则认为这两个词联系非常紧密。对于文档集合，可以将文档按顺编号（i=0…n-1），将文档编导作为向量索引，这样就有一个n维的向量。当一个词出现在某个文档i中时，向量i处值为1，这样就可以通过一个类似[0，1，0，…，1，0]形式的向量表示一个词。基于文档的词向量能够很好的表示词之间的关系，但是向量的长度和语料库的大小相关，同样会存在维度变化问题。

考虑一个固定窗口大小的文本片段来解决维度变化问题，如果在这样的片段中，两个词出现了，就认为这两个词有关。举例来讲，有以下三句话： “我\喜欢\你”，“我\爱\运动”，“我\爱\摄影”，如果考虑窗口的大小为1，也就是认为一个词只和它前面和后面的词有关，通过统计共现次数，我们能够得到下面的矩阵：



可以看到这是一个n\*n的对称矩阵X，这个矩阵的维数会随着词典数量的增加而增大，通过SVD（Singular Value Decomposition，奇异值分解），我们可以将矩阵维度降低，但仍存在一些问题： 矩阵X维度经常改变，并且由于大部分词并不是共现而导致的稀疏性，矩阵维度过高计算复杂度高等问题。

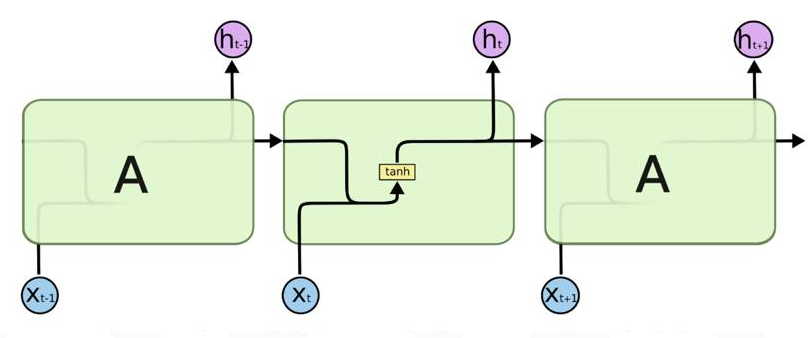
Word2vec是一个多层的神经网络，同样可以将词向量化。在Word2vec中最重要的两个模型是CBOW（Continuous Bag-of-Word）模型和Skip-gram（Continuous Skip-gram）模型，两个模型都包含三层： 输入层，投影层，输出层。CBOW模型的作用是已知当前词Wt的上下文环境（Wt-2，Wt-1，Wt+1，Wt+2）来预测当前词，Skip-gram模型的作用是根据当前词Wt来预测上下文（Wt-2，Wt-1，Wt+1，Wt+2）。在模型求解中，和一般的机器学习方法类似，也是定义不同的损失函数，使用梯度下降法寻找最优值。Word2vec模型求解中，使用了Hierarchical Softmax方法和NegativeSampling两种方法。通过使用Word2vec，我们可以方便的将词转化成向量表示，让计算机和理解图像中的每个点一样，数字化词的表现。

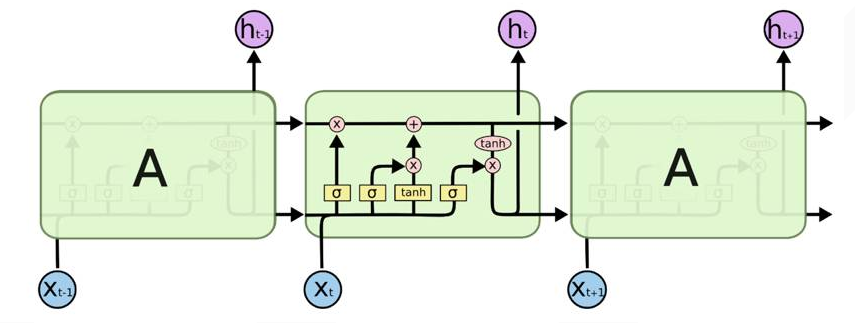
## 4.3 LSTM模型介绍

深度学习有很多种不同类型的网络，在图像识别领域，CNN（Convolutional Neural Network，卷积神经网络）使用的较多，而在NLP领域，考虑到上下文的RNN（Recurrent Neural Networks，循环神经网络）取得了巨大的成功。在传统的神经网络中，从输入层到隐藏层到输出层，层之间是全连接的，但是每层内部的节点之间是无连接的。因为这样的原因，传统的神经网络不能利用上下文关系， 而在自然语言处理中，上下文关系非常重要，一个句子中前后词并不独立，不同的组合会有不同的意义，比如”优秀”这个词，如果前面是”不”字，则意义完全相反。RNN则考虑到网络前一时刻的输出对当前输出的影响，将隐藏层内部的节点也连接起来，即当前时刻一个节点的输入除了上一层的输出外，还包括上一时刻隐藏层的输出。RNN在理论上可以储存任意长度的转态序列，但是在不同的场景中这个长度可能不同。比如在词的预测例子中： 1，“他是亿万富翁，他很？”; 2，“他的房子每平米物业费40元，并且像这样的房子他有十几套，他很？”。从这两个句子中我们已经能猜到?代表“有钱”或其他类似的词汇，但是明显，第一句话预测最后一个词时的上线文序列很短，而第二段话较长。如果预测一个词汇需要较长的上下文，随着这个距离的增长，RNN将很难学到这些长距离的信息依赖，虽然这对我们人类相对容易。在实践中，已被证明使用最广泛的模型是LSTM（Long Short-Term Memory，长短时记忆）很好的解决了这个问题。

LSTM最早由Hochreiter及 Schmidhuber在1997年的论文中提出。首先LSTM也是一种RNN，不同的是LSTM能够学会远距离的上下文依赖，能够存储较远距离上下文对当前时间节点的影响。

所有的RNN都有一串重复的神经网络模块。对于标准的RNN，这个模块都比较简单，比如使用单独的tanh层。LSTM拥有类似的结构，但是不同的是，LSTM的每个模块拥有更复杂的神经网络结构：4层相互影响的神经网络。在LSTM每个单元中，因为门结构的存在，对于每个单元的转态，使得LSTM拥有增加或减少信息的能力。



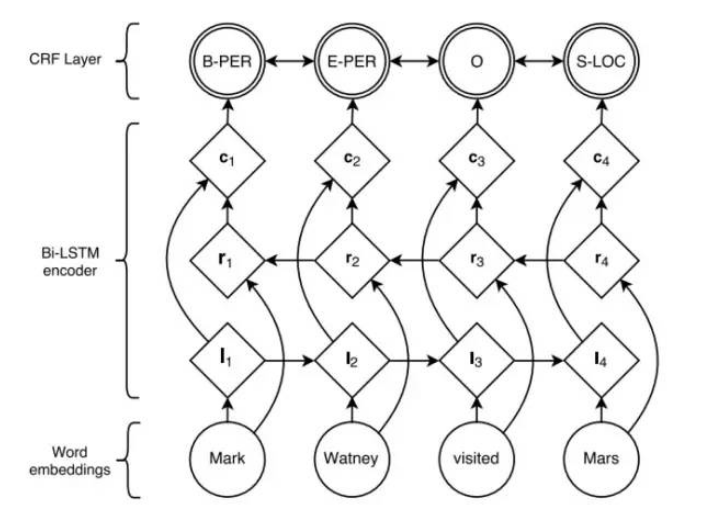


## 4.4使用深度学习的中文分词实例

深度学习主要是特征学习，端到端训练，适合有大量语料的场景。另外各种工具越来越完善，利用GPU可大幅提高训练速度。

前文提过，深度学习主要是特征学习，在NLP里各种词嵌入是一种有效的特征学习。本文实现的第一步也是对语料进行处理，使用word2vec对语料的字进行嵌入，每个字特征为50维。

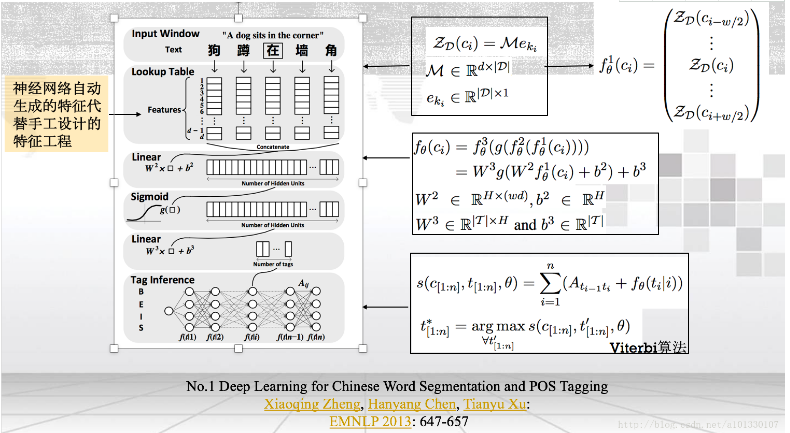
得到字嵌入后，用字嵌入特征喂给双向LSTM， 对输出的隐层加一个线性层，然后加一个CRF就得到本文实现的模型。



另外，字符嵌入的表示可以是纯预训练的，但也可以在训练模型的时候再fine-tune,一般而言后者效果更好。

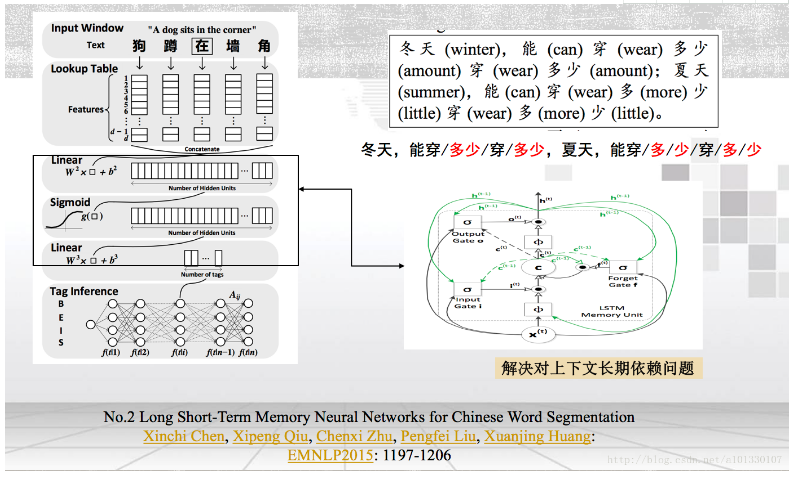
对于fine-tune的情形，可以在字符嵌入后，输入双向LSTM之前加入dropout进一步提升模型效果。

最后，对于最优化方法，文本语言模型类的貌似Adam效果更好， 对于分类之类的，貌似AdaDelta效果更好。



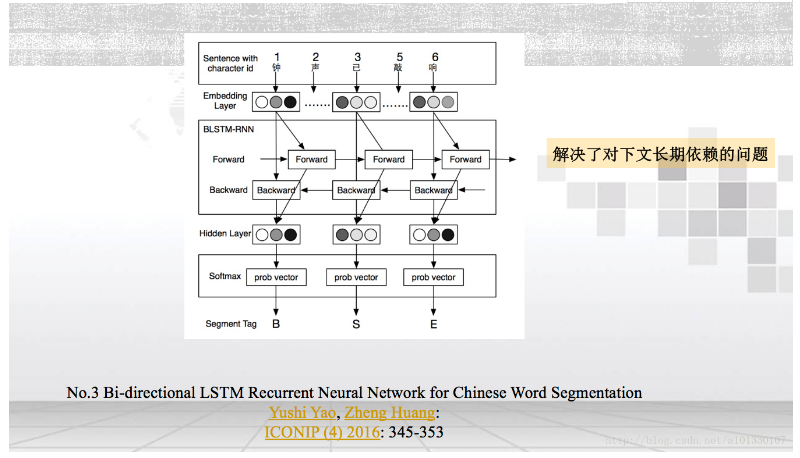
这篇作者是第一批使用神经网络来做中文分词和词性标注的，作者通过深层网络来自动获取任务相关的特征，从而避免使用了特定任务或是手工设计的特征工程，这种方法利用大规模非标注数据来改善中文字符的内在表示，然后使用这些改善后的表示来提高有监督的分词模型和词性标注模型的性能。本次的分词方式本质上是对输入的句子中的字进行标注。对于输入句子中的每一个字，神经网络架构将为其可能的每一个TAG进行评分，为了解决不同句子对应的字序列长短不一的问题，本文中采用的是窗口方法。窗口方法假定一个字的tag主要依赖于与其相邻的字。

接下来给大家讲一下这个模型的结构：首先通过lookup层得到窗口中每个字的字向量。之后将每个固定长度的字向量首尾相连得到一个新的特征。接下来的3层为一个标准的神经网络层。这个神经网络层的输出是一个包含每个标签得分的矩阵。最后使用Viterbi算法完成标注的推断。这个模型虽然具有不错的分词效果，但是它只利用了固定窗口大小的上下文信息，而忽略了一些长距离的文本信息。

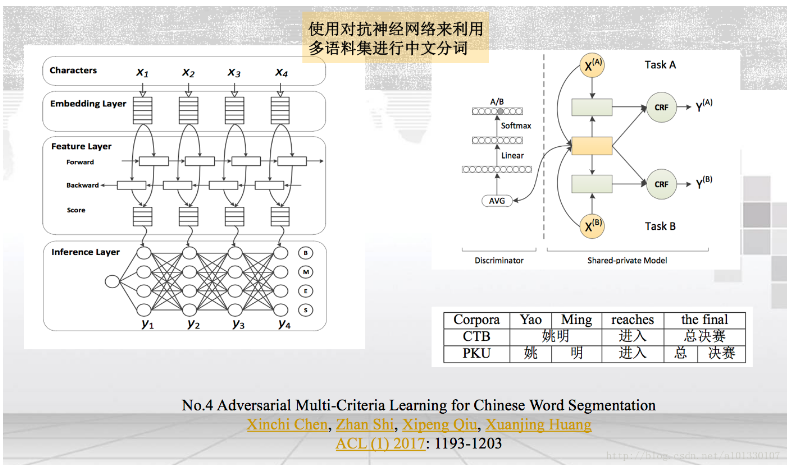


这篇文章的作者使用了LSTM来代替神经网络的隐层。改进了传统神经网络不能解决的长期依存关系的问题。作者认为上下文对分词很重要，比如右上图的例子，冬天，能穿多少穿多少，夏天，能穿多少穿多少。‘多少’这两个字是要通过‘冬天’，‘夏天’来协助分词的。但是按照之前那个方法，冬天和夏天两个词，或是类似这样的情况中的关键词，是不在窗口范围内，这样就会影响到分词的效果。所以把上下文考虑进来增强分词的性能。作者采用的方法就是将上篇文章中3层普通的神经网络层改为LSTM。使用输入门，输出门，遗忘门来决定如何利用和更新信息，明确地利用上文中的特征进行建模。所以，当LSTM的一个单元探测出比较重要的信息时，LSTM的单元能将这个信息一直保留到距离较远的下文中。比如例子中的，可以将冬天这个词带到多少这个词上，一同做推断。

但是LSTM只有上文，但作者认为上下文对分词效果都有影响。所以在这一点上，这个方式是存在缺陷的。虽然文中引入窗口来带入下文，但是不能获取长距离的上文。此外文中对中文习语的替换也存在争议。很多学者认为这样的做法是不可取的。后来有学者使用这篇文的方法，但未对习语进行替换，实验结果在每个数据集上确实都有所降低。



为了解决上文不能充分利用下文信息来分词的问题，此文提出使用了双向LSTM。这样可以充分的利用上下文。输入的字向量是D维的，但是经过双层LSTM，向量会变成是2D维，最后通过压缩，仍然输出为D维向量。由于用来中文分词的网络结构越来越复杂，模型的参数也越来越多，所以文中再训练过程中使用dropout来防止过拟合。



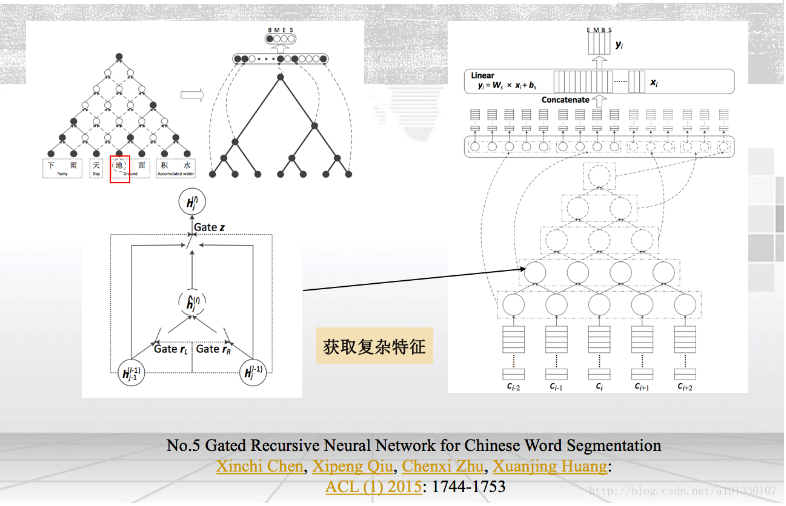
以往的分词方法是针对一个预料用一个种分割标准进行分词，但是在一般情况下一个预料遵循一种分割标准。比如在两个语料对句子‘姚明进入总决赛’的分割标准就不同。所以此文作者提出同时利用多个分词标准的语料集来进行分词。如何利用多个分词标准的语料集是这种分词方法的一个重要组成部分。此文同时用八个预料用八种分词标准对句子进行分词，并在八份语料中都提升了准确率。在多份语料上的模型训练可以堪称一个多任务学习问题。这些任务之间显然存在很大的共性，但也存在一些差异。文中具体采取的方案是拿出8个特定于具体任务的私有LSTM模块，跟原来共享的LSTM网络模块一同构成左图的特征抽取层，这一块的具体的结构如右图。两个灰色的私有LSTM模型负责捕捉TaskA和TaskB的任务私有特征，中间黄色的共享LSTM模块负责捕捉任务共享特征，然后再把私有特征与共享特征拼接到一起，输入每个任务私有的CRF模块。

如果共享特征中混入了任务私有特征时，那么会导致分词准确率下降，所以文中利用了对抗网络来将私有特征从共享的LSTM模块中剥离出去。所以作者加了一个判别器，就是右图中的左半部分。这个判别器的主要负责检查共享特征中是否混入了特定于某个任务的特征。

具体来说，每当一个样本经过字向量层、共享 LSTM 层之后，我们会得到一个特征向量序列，该序列的长度与输入样本的字符长度相等。为了检查这些特征向量是否“纯净”，所以对它们求平均，得到一个固定长度的特征向量，再输入判别器网络模块，要求判别器预测该特征向量来源于 8 个语料中的哪一个。这就是判别器设定的目标。

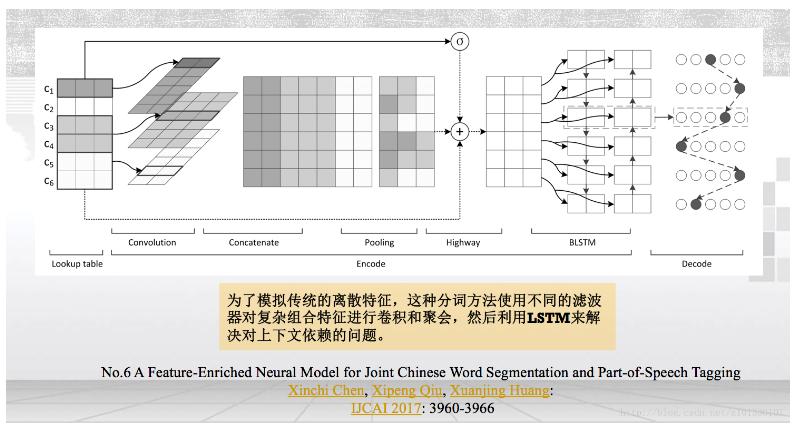
假如判别器能够准确预测每一个共享特征向量的来源语料，则说明这些共享特征中混入了太多私有信息。所以给共享 LSTM 模块设定一个目标是，让它跟判别器对抗，想办法让判别器预测不准。假如共享 LSTM 模块成功让判别器分不清特征向量来自哪个语料，意味着已经把私有特征剥离出去了，从而保证了共享特征向量的纯净性。

这样的做法既能充分享受到数据量增大带来的好处，又避免了不同语料之间相互掣肘，做到了“求同存异，共创双赢”。



之前的神经网络模型非常专注于其最小化特征工程对模型的影响，通过神经网络自动生成句子的特征。之前的做法只是将字符向量连接起来作为输入，这样的拼接方式过于简单，不能像传统方法一样提取出复杂的特征。所以此文的作者提出了Gate Recursive神经网络的方法，它包含reset和update门以融合上下文字符的复杂组合。此文采用分层的训练方法，避免了梯度扩散问题，使用了dropout策略避免了过度拟合问题。

如左上图所示，在给定上下文的情况下，为了预测字符‘地’的标签，本文使用有向无回路图来结合左右子节点的信息给父节点，如右图所示。GRNN从底层到顶端递归地检测组合，通过reset和update门来判断神经元是活跃的，还是被抑制的，其中固体节点表示活跃的神经元，而空心的则表示被抑制的神经元。每一个神经元都可以看作是一个复杂的特征。最后通过把所有组合信息输入到网络后，得到标签的分数向量。GRNN可以通过重置门和更新门来选择和保存有用的组合。这些组合和传统方法中的离散的特征工程起着同样的作用。但是它们之间的区别在于，神经系统会自动地学习复杂的组合，而传统的方法则需要人工设计它们。



这个模型是另外一个引入复杂特征的CNN-LSTM的网络模型，这种方法的目的是在不增加模型复杂性的基础上提高对上下文长距离的依赖并提取出相对复杂的特征进行建模。

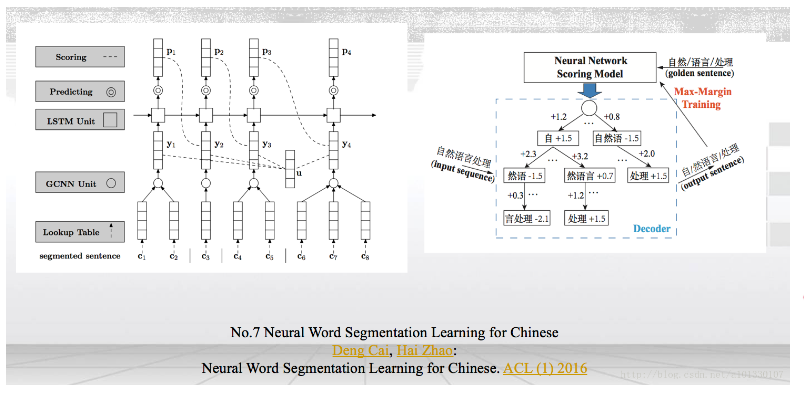
在模型中有几个关键部分：

1.用卷积层来模拟组合特征，这些特征可以代替复杂的人工定义的特征：这个卷积层为每个字符模拟了n-gram特征。每个字符的特征就是所有不同特征映射集合对应列的连接。

2.用一个Pooling层来挑选出最价值的特征

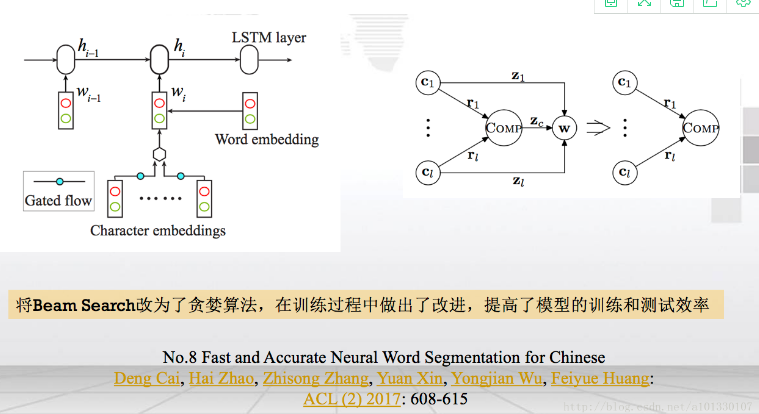
3.加入一个highway层，在提高模型的深度，获取更显著的特征。同时可以加快模型收敛速度，减轻梯度消失的问题。

4.用一个双向LSTM层来获取上下文的信息。

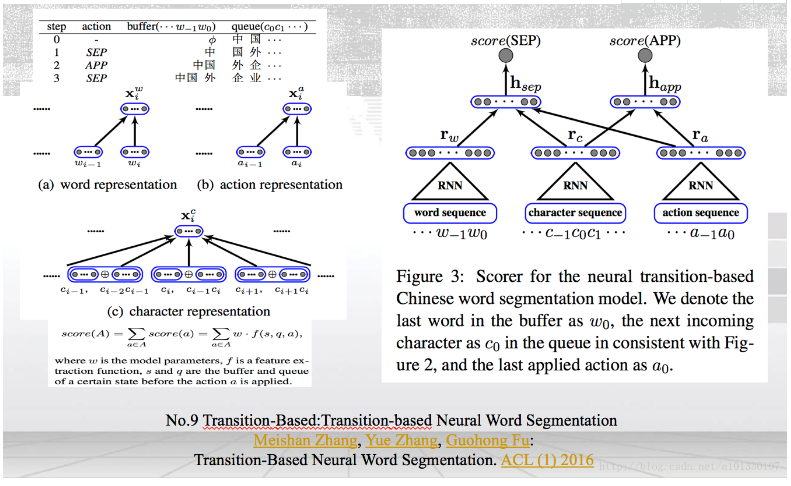


这种分词方式的本质就是对句子中的字进行标注。接下来我将给大家介绍一些基于词的分词方法。此篇文章提出来一种新的神经网络框架，它彻底地消除了上下文的窗口，它是利用完整的分割历史，直接对分词结果建模。分词结果的好坏主要由每个单词成词的分数和单词连续的分数来衡量。

此方法的框架是用一个神经网络网络评分模型来评估分词方式的可能性。其结构如左图所示。第一层通过映射得到每个字的特征，第二层是枚举定长L=4之内的所有单词，用GCNN组合，得到词语向量。之后将其与一个参数向量内积得到词语分数。第三层用LSTM将分词历史集合起来。其中分词结果的得分就是单词得分与句子得分之和。在这个评分模型基础上，建立一个译码器，如右图所示，用来获得最高分数的分词句子。



这篇文章基于上篇文章做了改进，将其中的Beam search改为了贪婪算法，在训练过程中采用了两种策略：Early Update和LaSO Update。提高了模型训练和测试的效率。此外在字向量转为词向量的部分也做出了改进，作者列出了一个包含最有可能连续的词列表，如果词在列表中则链接到平均池，如果不在列表中，则单独处理。



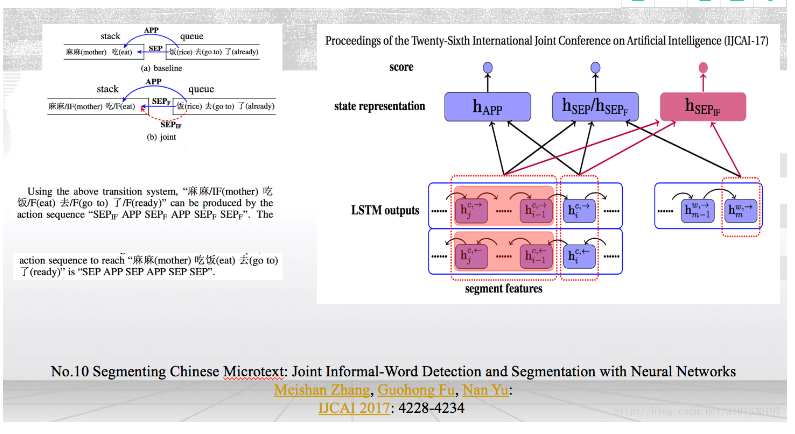
此篇文章也是基于词的中文分词。文本使用Transition-Based模型从左到右，递增地对句子进行解码，使用字符级和词级的特征来给分词结果进行打分。此文使用卷积神经网络实现局部特征组合，使用LSTM分别地学习全局语句级特征，这个模型可以利用丰富的嵌入特性。

左图是此文的baseline，它使用Transition system递增地去分词。它把已经做过部分分词的句子放在buffer s中，将接下来要进行分词句子中的字符放在quene q中。其中action包括Append（APP），Separate（SEP）。APP代表从序列 q中除去第一个字符，把这个字符加到buffer中最后一个词的后面。SEP代表将第一个字符从序列中去除掉，把它放在buffer中作为一个单独的词。举个例子：中国外企业务发展迅速。基于这个框架，使用一个编码器来寻找一组最优的action sequence。这个框架中的基本的特征包含三方面的信息。这三方面的信息后面也用到了作者的模型中，第一个信息是序列q中第一个字符和buffer中的最后一个字符用来给SEP和APP动作来打分。第二个信息是通过已经被识别的词来指导SEPaction。第三个信息是已识别的词的相关信息，比如它们的长度，这个词中的第一给字符或是最后一个字符可以作为额外的特征。

右图就是此文的方法，如图所示，作者使用的方式是使用CNN，并且窗口为2的方式来获取到词特征向量。同样也是通过使用CNN来获取动作特征向量。使用BI-LSTM来获取字符特征向量。

根据得到三个不同的向量作为输入向量，输入到LSTM中计算Rw，Rc，Ra。

作者利用字符特征Rc，词特征Rw，动作特征Ra这三个特征来给SEP动作打分，使用字符特征Rc和动作特征Rw来给APP动作打分。



现在先进的中文分词系统在手动标注语料库上训练得出的模型，在类似的标准测试语料库中正确率可以达到95%以上。但是当这些模型运用于像微信、微博、SMS这类数据时，正确率会下降很多。因为之前的模型都是使用标准语料库进行训练的，它们的词都是正规的，但像微博一类的数据中，包含很多不正规的用词，比如，人艰不拆，累感不爱。这样的词会大大降低模型的性能。本文在transition-based的神经网络模型的基础上增加对非正规词的处理。对于非正规词，此文的处理方法是1.我们在外部字典的帮助下，自动构建训练示例。2.将规范化作为未来的一个步骤，使我们的模型能够通过低质量的字典检测非正式词汇。

此文是在上文的基础上进行改进的，之前转移动作只有：APP，SEP。次文针对于非正规词，增加一个转移动作SEPif，就是当非正规词以一个与前文分开的单个词进入stack时，转移动作标为SEPif。在transition-based的神经网络模型的改进时在state representation层增加一个隐层来处理非正规词。目标函数从Max-margin改为max-entropy. 因为作者通过实验发现Max-margin方法对对模型参数的初始化比较敏感。

# 参考文献

[1]龚汉明, 周长胜. 汉语分词技术综述[J]. 北京信息科技大学学报(自然科学版), 2004, 19(3):52-55.

[2] 朱德熙.语法讲义[M] . 北京: 商务印书馆.1982.

[3] 张春霞,郝天永.汉语自动分词的研究现状与困难[J].系统仿真学报.2005.(17）.138-147.

[4] 李家福，张亚非． 基于EM算法的汉语自动分词方法[J]． 情报学报，2002，21(3)，P269-272．

[5] 黄昌宁. 中文信息处理中的分词问题[ J] . 语言文字应用,1997,( 1): 72-78.

[6] 谭琼，史忠植． 分词中的歧义处理[J]．计算机工程与应用，2002，38(11)，P125-127．

[7] 张启宇, 朱玲, 张雅萍. 中文分词算法研究综述[J]. 情报探索, 2008(11):53-56.