面向领域快速移植的高精度汉语分词系统研究

匡 盟 盟

院 （系）：计算机科学与技术　　专　　业：计算机科学与技术

学　　号：1143220116　　指导教师：赵铁军 教授

2018 年 **7** 月

哈爾濱工業大學

毕业设计（论文）

题　　目　面向领域快速移植的高精度汉语分词系统研究

专　　业　计算机科学与技术

学　　号　1143220116

学　　生　匡盟盟

指 导 教 师　赵铁军　教授

答 辩 日 期　2018/06/13

摘　要

汉语分词就是将中文语句中的词汇按照使用时的含义切分出来的过程，也就是将一个汉字序列切分成一个个有单独含义的词语。

汉语自动分词是自然语言处理领域的基础性、关键性的任务，其准确率直接关系到上层的高级算法的实施以及其效率。随着几十年的相关领域算法的发展，在通用领域文本上，较好的分词算法已经能够达到很好的结果，但是一般的分词器向不同特定领域文本移植的时候，又会产生因专业领域新词不断出现、词典不全或者完善难度大等而出现准确率下降的问题。本文主要阐述一个以条件随机场（CRF）模型为核心的高精度分词系统以及一个用于面向领域（主要适用于医疗、法律与金融三个领域）快速移植的该分词系统的部件。

本文工作主要包括这样两个部分：第一部分，在已经取得很好分词效果的 CRF 模型分词基础上，通过增加频繁串统计、字符规范化、额外词典等前处理以及模型自学习、用户自定义词典等后处理手段进一步提高模型分词准确度。第二部分，在已有高精度分词系统的基础上，通过领域词典、规则匹配、感知机命名实体识别将句子中可能存在的特定领域的专有名词识别出来并整合分词结果，形成最终的面向领域快速移植的分词系统。

本工作主要创新点包括：首先，完善领域词典（医疗、法律与金融三个领域），并将其用作特定领域分词任务；其次，制定了特定领域词汇的匹配规则，结合领域词典，可以达到比较好的识别效果；最后，通过与机器学习基础算法——感知机相结合，通过识别特定领域命名实体来达到进一步提高特定领域分词准确性的目的。

关键词：领域移植；汉语分词；CRF；领域词典；规则匹配；感知机

Abstract

The Chinese word segmentation is the process of segmenting the Chinese vocabulary according to its meaning when used. It is to divide a Chinese character sequence into words that have separate meanings.

Automatic word segmentation in Chinese is a fundamental and critical task in the field of natural language processing. Its accuracy is directly related to the implementation of advanced algorithms and its efficiency. With the development of algorithms in related fields for several decades, better word segmentation algorithms have been able to achieve very good results in general field texts, but when general word segmenters are transplanted to texts in different specific fields, they will produce The emergence of new words in the field, incomplete dictionaries, or difficulty in perfection, etc., has led to the problem of a decrease in accuracy. This article mainly describes a high-precision word segmentation system centered on the conditional random field (CRF) model and a component of the word segmentation system for field-oriented (mainly applicable to medical, legal and financial areas) transplantation.

The work of this article mainly includes these two parts: In the first part, based on the CRF model word segmentation that has achieved a good word segmentation effect, the preprocessing such as frequent string statistics, character normalization, additional dictionaries, model self-learning, user-defined dictionaries, etc. are increased. Post-processing means further improve model segmentation accuracy. In the second part, on the basis of the existing high-precision word segmentation system, domain name dictionary, rule matching, and perceptron machine named entity recognition are used to identify the specific domain-specific nouns in the sentence and integrate the word segmentation results to form the final orientation. Field fast word segmentation system.

The main innovations of this work include: First, improve the domain dictionary (medical, legal and financial fields) and use it as a word segmentation task in a specific field; secondly, formulate matching rules for specific areas of vocabulary, combined with domain dictionaries, can achieve A better recognition effect; Finally, through the combination of a machine learning basic algorithm, a perceptron, and the identification of domain-specific entities, the purpose of further improving the accuracy of word segmentation in a specific domain can be achieved.

Keywords: Domain Transplantation, Chinese Word Segmentation, CRF, Domain Dictionary, Rule Matching, Perceptron

目　录

摘　要..... I

ABSTRACT..... II

目　录..... III

第 1 章 绪论..... 1

1.1 选题的背景及意义..... 1

1.2 汉语分词难点..... 1

1.3 国内外汉语分词研究现状..... 2

1.3.1 国内研究..... 2

1.3.2 国外研究..... 3

1.4 领域移植汉语分词研究现状..... 4

1.5 汉语分词的评测方法..... 4

1.6 主流开源分词器使用算法与结果..... 5

1.7 其他分词算法..... 6

1.7.1 基于词典的分词算法..... 6

1.7.2 基于统计的分词算法..... 7

1.7.3 基于深度学习的分词算法..... 7

1.8 研究内容及论文的组织结构..... 8

第 2 章 高精度汉语分词系统设计与实现..... 9

2.1 高精度汉语分词..... 9

2.1.1 高精度分词简介..... 9

2.1.2 主要瓶颈..... 9

2.1.3 解决方案..... 9

2.2 条件随机场..... 10

2.2.1 CRF 原理..... 10

2.2.2 概率计算..... 11

2.2.3 学习算法..... 12

2.2.4 特征选择..... 13

2.2.5 预测算法..... 13
2.3 前处理与后处理..... 14
2.3.1 前处理..... 14
2.3.2 后处理..... 15
2.4 分词过程..... 17
2.5 高精度分词器性能..... 18
2.5.1 测试前处理性能..... 18
2.5.2 测试后处理性能..... 18
2.5.3 整体性能..... 20
2.6 本章小结..... 21
第 3 章 面向领域移植的汉语分词系统设计与实现..... 22
3.1 领域移植的汉语分词..... 22
3.1.1 基本介绍..... 22
3.1.2 主要问题..... 22
3.1.3 解决方案..... 22
3.1 领域词典..... 23
3.2 启发式规则..... 23
3.3 感知机辅助分词..... 23
3.4 快速移植..... 24
3.5 系统整合..... 25
3.5 领域移植分词系统性能..... 25
3.6 本章小结..... 26
第 4 章 评测资源与评测方法..... 27
4.1 评测资源..... 27
4.2 评测方法..... 27
4.3 本系统测试结果..... 27
4.4 对比测试结果..... 28
4.5 速度测试..... 29
4.6 结果分析..... 29
结 论..... 30

参考文献..... 32

哈尔滨工业大学本科毕业设计（论文）原创性声明..... 35

致　谢..... 36

第 1 章 绪论

1.1 选题的背景及意义

自然语言处理是在当今的“智能计算”时代，让机器实现“能看会想、能听会讲”的首要任务，而汉语分词又是自然语言处理任务中最基础、最重要的一部分。有别于外文，汉语句子没有词的界线，因此在对汉语做自然语言处理时，通常需要先自动分词，分词的效果将直接影响后续的诸如命名实体识别、关系抽取等操作的效果。分词这个基础性工具，随着场景的不同，要求也自然不同。例如我们每天都在用搜索引擎，评价其好坏的主要标准就是搜索结果的相关度排序，而汉语分词的准确与否直接影响了这一排序过程，无疑在这一方面准确的汉语分词系统意义重大。

对于中文而言，词是承载语义的最小单元，由词构成语句，又由语句构成篇章。但是，中文文本是由连续的字序列构成，中文和很多外文一个重要区别就是：外文多是通过空格来分隔一个词，而中文词与词之间是没有天然的分隔符，这就造成了中文处理上的难度。汉语自动分词，就是通过计算机自动地完成分词任务，来解决这一问题。经过几十年的中文自动分词研究，在一般文本上的分词技术已经相当成熟，而当面临向不同领域移植时，必然会由于不同领域的文本内容的变化带来诸多训练语料中未出现的领域词汇，使得未登录词识别、歧义切分问题成为跨领域分词的一个关键问题。

1.2 汉语分词难点

对于一般通用文本的汉语分词存在的难点，主要有以下两个方面：

1.未登录词：也被称为“新词”，即特定的专有名词，例如：机构名、地名、人名、物品名、商标、简称、缩写等。

2.切分歧义：即对同一个原始字符串的多个分词可能结果。切分歧义主要包括交集型歧义、组合型歧义以及真歧义。例如：“从小学电脑”，可能切分成“从|小学|电脑”，也可能分成“从小|学电脑”，这种称为交集型歧义（交叉歧义）。分词有不同的粒度，指一个短语可以切分开，也可以不切开，如“中华人民共和国”，

可以作为一个词，也可以切分成“中华|人民|共和国”三个词，这又是另一中切分歧义——组合型歧义。最后一个歧义难题——真歧义。真歧义是给出一句话，即使是人也不好判断哪个是词，该在哪里切分，哪里不切。

在中文分词技术快速发展到达一定的极限的今天，通用领域或者同一领域训练、测试的方法已经能够取得很好的效果，但是当面临通用领域训练向特定领域迁移测试或者测试语料与训练语料领域不一致时，分词准确率明显降低。同时，为每个不同的特定领域都标注足够的训练样本是一件基本不可能实现的事，其人力物力消耗是极大的，我们能收集到的只有通用领域的一些标注结果。

在领域移植分词任务中，由于不同的特定领域文本在内容上、特征信息上以及文本上下文上有很多不同，所以会导致未登录词识别、歧义切分等方面有更大的挑战。

1.3 国内外汉语分词研究现状

1.3.1 国内研究

国内研究汉语分词的科研单位主要有：中科院、清华、北大、北京语言学院、东北大学、MSRA、IBM 研究院以及哈工大等。

国内主要的成熟的分词系统：ICTCLAS（汉语词法分析系统）、海量信息、盘古分词、结巴分词、BosonNLP、清华 THULAC 以及哈工大语言云（LTP-Cloud）等。

国内在汉语分词算法的研究上进展颇丰，参与的科研机构也比较多，使用的方法也比较多，从文献[1]—[19]可以看出。国内分词算法上的进展主要有：2005 年，文献[13]中哈尔滨工业大学在分词阶段以聚焦于基于词的 n-gram 方式。算法流程大致为：先将词依照已有词典进行初步切分，同时从训练语料中统计得出 3-gram 信息，动态规划确定哪条切分路径为最优路径从而确定分词位置。2007 年，文献[19]中赵海等人研究了基于子串标注的分词算法，其在 Bakeoff-2005 测试集上准确度较高。2009 年，文献[3] 使用基于 N-gram 的自动中文分词系统，它结合了分词和词性标注，并且使用词性标注结果来参与评估分词结果。文献[34]又提出了一种字词联合解码的分词技术，方法中使用了字、词信息，其充分发挥由字构词在识别未登录词方面的非凡能力。2010 年，文献[35] 提出了一种基于文字分类边界的分词算法，以文字边界为分类判据，判断是否为单词分词，从而达到分词的目的。文献[36]将基于字的生成模型与基于字的判别模型进行联合用于分词任务中。2014 年，

文献[29]对文献[28]的模型做了十分重要的改进，并引入了标签向量以更精细地刻画标签之间转移的关系，其改进效果类似于向 ME 模型之中引入 Markov 特征。2015 年，为了更完整精细地对分词上下文进行建模，文献[30]提出了一种带有自适应门结构的递归神经网络（GRNN）抽取词的 n-gram 特征，其中的两种定制的门结构（分别为重置门和更新门）被用来控制 n-gram 信息的融合和抽取。2016 年，文献[31]中递归神经网络（GRNN）和长期和短期记忆网络模型（LSTM）被结合用于分词任务。在该模型中，其先通过 LSTM 提取上下文的敏感的局部信息，然后通过滑动窗口将这些局部信息用带门结构的递归神经网络（ANN）融合起来，最后用作标签分类的依据来进行分词。文献[32]提出了一种基于词之间转移几率的模型用于分词任务，并将传统的特征模版与神经网络主动学习的特征相结合，在神经网络自动学习的特征和传统的离散特性的交融方面做了尝试。2017 年，文献[33]通过简化网络的结构，混合字、词输入以及使用早期更新（early update）等收敛性更好的训练策略，设计出了一种基于贪心搜索的快速分词系统。该算法与之前的深度学习算法相比不仅在速度上有了巨大提升，分词精度也得到了很大程度的提高。

1.3.2 国外研究

国外研究汉语分词的主要科研机构有：斯坦福、SUTD、UC Berkeley、CMU、CityU 等。

国外成熟的分词系统有：Core NLP（斯坦福 NLP Group）、Zpar（SUTD）、Basis Technology、Open NLP (Apache 基金会)等。

国外分词算法上的进展：2003 年之前，研究重点主要集中在词典与人工规则相结合、词典与概率统计规则相结合的分词算法上。从 2005 年开始，逐渐使用基于字序列标注的分词算法，该算法开始于文献[20]，第一次将严格的串标注学习方法应用于分词。在文献[21]和文献[22]之后，文献[23]与文献[24]出现，使基于 CRF 模型的分词崭露头角，在此之后，CRF 多个变种甚至构成了深度学习时代之前的标准分词模型。不久，一种基于词的随机过程模型产生了一个 CRF 变种，即 semi-CRF(半条件随机场)模型并直接应用于分词相关的研究。2006 年，基于字序列标注的分词方法已经开始大规模盛行，此时，分词使用的核心模型仍然是 ME 和 CRF，同年，文献[25]揭晓了利用 semi-CRF 实现的第一个分词器。文献[26]提出了一种基于子词（subword）的标注学习算法，其基本思路：从训练集中抽取高频的已知词构造成子词词典。2007 年，ME 的方法已经开始退出分词舞台，CRF 越来越成为

主流模型。2010 年，分词上的核心方法仍然是基于 CRF 模型，后处理的方法一般使用的是 SVM-HMM 联合模型。2011 年，当子串的抽取和统计度量得分计算扩展到训练集之外，文献[27]实际上提出了一种扩展性很强的半监督学习分词算法，实验也验证了该算法的有效性。2013 年，文献[28]提出神经网络汉语分词方法，首次验证了深度学习方法应用到汉语分词任务上的可行性。

1.4 领域移植汉语分词研究现状

在领域自适应方面相关研究比较少，2008 年，文献[45]所述的方法利用单个汉字的单词形成能力和单词形成模式，计算单词形成技能和单词形成模式。作为发现新词的规则，在科技领域的文本中做了新词发现和新模式发现实验，取得非常好的结果。2012 年，文献[41]中提到通过将外部字典信息合并到统计分词模型（使用 CRF 统计模型）来增强域适应。在确定一个领域并给出这个领域的文献数据集合的前提下，文献[44] 主要从以下两个步骤进行新词的发现：1.针对特定领域的文字分词处理收集文档，采用统计 n-gram 方法进行分词，更有效地找到字典中不存在的新词（未登录词）; 2 新的专业领域词汇提取。这是一个基于现有专业词汇发现未知专业词汇的过程。 目的是从上一步中生成的新词汇中提取新词。 属于目标域的专业词汇，在这一步使用 Apriori 方法。2013 年，文献[40]实现了基于生语料的领域自适应分词模型和双语引导的汉语分词模型，并提出融合多种分词结果的方法，通过构建格状（Lattice）结构并使用动态规划算法得到最优汉语分词结果。 此时，由耶鲁大学教授提出的 Active Learning（主动学习）得算法到了较为广泛的使用。2015 年，文献[39]结合主动学习和 n-gram 统计特征，提出了一种分词方法。通过对目标域文本和现有标记语言数据差异的统计分析，将包含最多未标记语言现象的小规模语料库优先用于手动标记。 这种方法已被证实可以提高科学文献中的分词效率。文献[43]中卡方统计和边界熵用于提高未登录词的处理能力，采用自学习和协作学习策略，进一步提高字词标注和分词方法在特定领域适应中的效率。2016 年，文献[42]又提出了一种结合 CRF 和领域字典的方法来提高领域适应性。根据单词形成规则，提出了三种消除歧义的方法：固定词串解析，动词解析和词概率解析。

1.5 汉语分词的评测方法

为了评测我们的分词器优劣，评测必不可少。为了设定一个有说服力的评测，我们在这里采用 Bakeoff 比赛的评测方法。整个评测工作可以分为三个部分：制定标准答案、设置评价指标以及计算方法。由于即使是人工来分词，对于大规模文本，都不一定会达成统一意见，所以制定标准答案就显得极其基础与重要，在后文中我将详细介绍我们使用的测试语料的标准答案。目前已经完全被大家接受的评价指标有：精确度（P）、召回率（R）与 F 值（F），精度表明了分词器分词的准确程度（P=正确数/分词结果总词数），召回率表明了分词器切分正确的比例（R=正确数/标准答案总词数），F 值综合反映整体的指标，为前两者的调和平均（F=2*P*R/(P+R)），这三个指标都是越大越好（最大值为 1，即 100%正确） 。

1.6 主流开源分词器使用算法与结果

在这里只基本介绍几种开源的使用较为广泛的分词器：

结巴分词在算法上采用的是由字成词的隐马尔科夫模型（HMM），利用动态规划切分。数据结构上利用前缀 Trie 树实现高效的词图扫描，得到句子中汉字所有可能成词情况所构成的有向无环图 (DAG)。

THULAC（THU Lexical Analyzer for Chinese） 是一个中文词法分析的工具程序，其由清华大学开发，具备中文分词和词性标注的功能。其训练样本为目前世界上规模最大的人工分词和词性标注的中文语料库（约含 5800 万字）。

哈工大 LTP（Language Technology Platform）的分词模块（2015 年前）基于结构化感知器（Structured Perceptron）算法、（现在）基于条件随机场（CRF）算法构建，准确度非常高、速度也比较快；同时支持用户自定义词典，适应不同用户的需求；另外还新增了个性化（增量式）训练功能，用户可以根据自己的实际需求，如对新领域的文本进行分词等。

这几个分词器官网给出的具体性能指标见表 1。

分词器	算法	P	R	F
结巴分词	HMM+Viterbi	0.850	0.784	0.816
THULAC	-	0.939	0.944	0.941
哈工大 LTP	CRF	0.972316	0.970354	0.972433

表 1 几种分词器的算法与指标

1.7 其他分词算法

1.7.1 基于词典的分词算法

最大匹配法以及其变形，应该是基于词典的方法的主角。从文献[46]中提到的苏联学者提出的算法：先建立一个最长词条字数为 6 的词典, 然后取句子前 6 个字查词典，如果查不到, 则去掉最后一个字继续查, 一直到找着一个词为止开始。最大匹配正式进入我们的视野，接着为了改良基本算法，自然而然地涌现出逆向最大匹配、双向最大匹配等算法。

文献[47]又提出一种复杂最大匹配法，其首次提出三词语块（three word chunks）的概念。在所有可能的三词语块中根据如下四条规则选出最终分词结果。

1：最大匹配 (Maximum Matching)

其核心的假设是：最可能的分词方案是使得三词语块(three-word chunk)最长

2：最大平均词长（Largest Average Word Length）

在句子的末尾，很可能得到的"三词语块"只有一个或两个词（其他位置补空），这时 1 就无法解决其歧义消解问题，因此引入规则 2，也就是从这些语块中找出平均词长最大的语块，并选取其第一词语作为正确的词语切分形式。这个规则的前提假设是：在句子中遇到多字词语的情况比单字词语更有可能。

3：最小词长方差（Smallest Variance of Word Lengths）

1 和 2 并不能解决所有的切分歧义。因此引入规则 3，也就是找出词长方差最小的语块，并选取其第一个词语作为正确的词语切分形式。在概率论和统计学中，随机变量的方差描述了它的离散程度。 因此，规则的前提是句子中单词的长度通常是均匀分布的。

4：最大单字词语语素自由度之和（Largest Sum of Degree of Morphemic Freedom of One-Character Words）

上述三个规则可能都无法解决某些歧义消解问题[1]。直观来讲，有些汉字很少作为词语出现，而另一些汉字则常常作为词语出现，从统计角度来看，在语料库中出现频率高的汉字就很可能是一个单字词语，反之可能性就小。

再后来对最大匹配算法的改进，就是基于词典和规则的。其优点是实现简单，算法运行速度快，缺点是严重依赖词典，无法很好的处理分词歧义和未登录词。因此，未登录词识别模块是该方法的难点所在。

[1]有可能两个"三词语块"拥有同样的长度、平均词长及方差。

1.7.2 基于统计的分词算法

自从薛念文等人在文献[20]、文献[48]中提到基于字标注的分词模型以来，汉语分词进入基于统计分词阶段。2003 年，薛在最大熵（ME, Maximum Entropy）模型上实现的基于字标注的分词系统参加了 Bakeoff-2003 的评测获得很好的成绩从而被广泛关注。

字标注质上是训练出一个字的分类器。模型框架如图 1 所示。

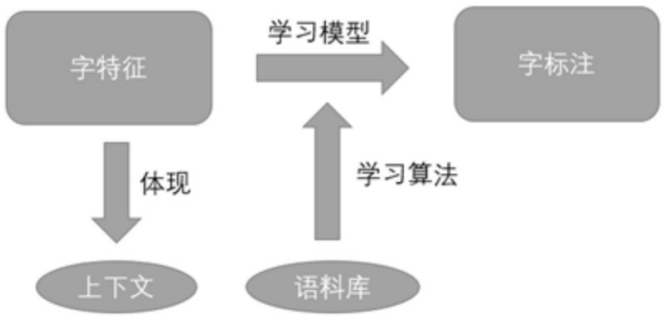


图 1 自标注模型分词

在此之后，基于隐马尔科夫模型、条件随机场等模型的分词逐渐登上历史舞台。

1.7.3 基于深度学习的分词算法

近几年来，随着深度学习方法的发展为分词技术带来了新的思路，直接以最基本的向量化原子特征作为输入，经过多层非线性变换，输出层就可以很好的预测当前字的标记或下一个动作。文献[50]中提到，在深度学习的框架下，仍然可以采用基于子序列标注的方式，或基于转移的方式，以及半马尔科夫条件随机场。深度学习主要有两点优势:

1. 深度学习可以通过优化最终目标，有效学习词的原子特征和上下文的表示；

2. 基于深层网络如 CNN、 RNN、 LSTM 等，深度学习可以更有效的刻画长距离的句子信息。

文献[51]中提出了一种深度学习框架，如图 2 所示，利用该框架可以进行中文分词。具体地，首先对语料的字进行嵌入，得到字嵌入后，将字嵌入特征输入给双向 LSTM，输出层输出深度学习所学习到的特征，并输入给 CRF 层，得到最终模型。

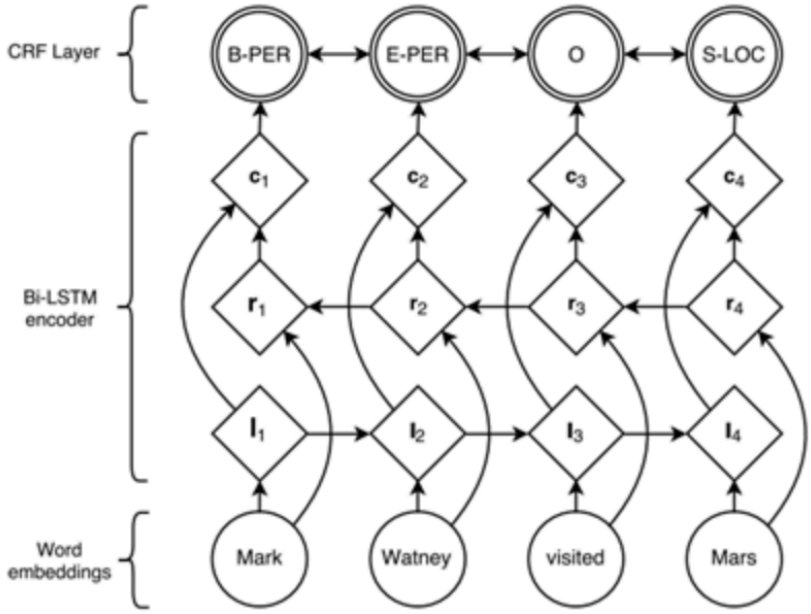


图 2 一种深度学习分词模型

1.8 研究内容及论文的组织结构

根据以上综述，我们选用已经在分词上取得巨大成功的 CRF 模型作为我们分词算法的基础模型，在此基础上我们通过前处理、后处理以及引入特定领域快速移植模块实现面向领域快速移植的高精度分词系统。

本文的组织结构大致如下：第二章介绍本系统中高精度汉语分词部分的设计与实现；第三章介绍本系统中面向领域移植的汉语分词部分的设计与实现；第四章通过实际的实验说明关于本系统的测试语料与测试结果；最后一章总结本次毕业设计结果并提出下一阶段工作。

第 2 章 高精度汉语分词系统设计与实现

2.1 高精度汉语分词

2.1.1 高精度分词简介

前文提到，词是承载中文语义的最小单元，无论篇章、段落还是句子，最基础、最原始的部分就是词。而且中文词与词之间是没有天然的分隔符，这就造成了中文处理上的难度。汉语自动分词，就是程序自动地完成分词任务，来解决这一问题。

高精度分词，顾名思义，就是分词准确度比较高的分词。由上一章所述，部分分词系统已经能达到 97%以上的准确度了。高精度汉语分词是更好地完成自然语言处理上层任务的基础，也是最重要的部分。在搜索引擎进行搜索时，首先就是要进行高精度分词，然后通过分出来的某些词去搜索索引进而返回结果，如果分词出现错误，不仅会加重搜索引擎的负担，而且得不到想要的结果。

2.1.2 主要瓶颈

在绪论中已经提到，一般分词任务都不可避免要解决未登录词、歧义切分、新词发现以及分词粒度四个主要的挑战。关于未登录词，这个是无法避免的，毕竟每时每刻都有新词出现，有的有一定规律，而有的则无迹可寻。歧义切分同样是一大难点。当我们识别出来了一个之前没见过的新串时，我们应该怎么做，这又涉及新词发现的任务了。还有一个比较迫切的问题就是分词结果的粒度大小的问题，例如“计算机科学与技术”，可以分为“计算机|科学|与|技术”，也可以分为“计算机科学|与|技术”，还可以作为一个整体为一个词，我们无法评判那种结果更对。

2.1.3 解决方案

本章介绍的分词方法属于基于统计的分词方法，在新词发现上有比较好的作用。同时在前处理中，增加了频繁串统计，基本可以有效地统计出频繁出现的词（一般可以发现新词）。为了处理未登录词问题，一方面通过自身模型的学习，另一方面可由用户自己输入（用户词典）。为了解决切分歧义，文本提出了一些前处理、后处理过程。由于本模型通过训练 1998 年人工标注的《人民日报》（细粒度）所

得，所以自然而然，本系统粒度较细，当然在后处理中适当将分散的词进行了合并，但也属于细粒度分词。

2.2 条件随机场

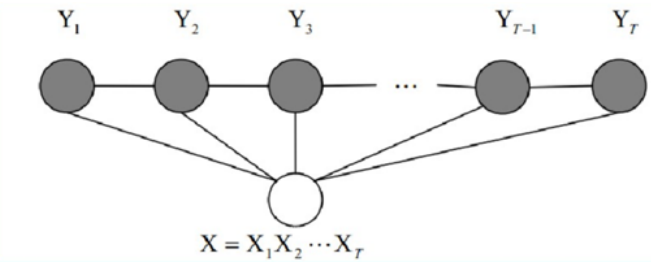
条件随机场（CRF）模型是由 J. Lafferty 等人于 2001 年提出，是一种用于标注和划分序列结构数据的概率化结构模型，在自然语言处理（NLP）领域得到了广泛应用。CRF 是一种判别式模型，即通过有限样本学习出判别函数（判别式）然后通过学习的结果去预测测试样本。一般如果说 CRF 序列建模，就专门指的是 CRF 线性链（linear chain CRF）。由于本次分词系统的模型训练部分采用 Java 版 CRF++，故下面简要介绍 CRF++采用的训练方法与处理手段。主要包括：CRF 原理、拟牛顿法确定目标函数、梯度下降、L2 正则化、L-BFGS 优化、前向-后向算法、特征选择以及维特比算法等。

2.2.1 CRF 原理

根据文献[42]，CRF 的原理大致可以表述成：设 G=(V, E) 为一个无向图，V 为结点集合，E 为无向边的集合，Y = { Yv | v∈V }，即 V 中每一个结点对应于一个随机变量 Yv, 其取值范围为可能的标记集合 {y}。如果以观察序列 X 为条件，每个随机变量 Yv 都满足以下马尔可夫特性：

$$p(Y_v \mid X, Y_w, w \neq v) = p(Y_v \mid X, Y_w, w \sim v)$$

其中，w~v 表示两个结点在图中为邻近结点。那么，(X, Y) 就是一个条件随机场。



在给定观察序列 X 时，某个特定标记序列 Y 的概率可以定义为：

$$\exp(\sum_j \lambda_j t_j(y_{i-1}, y_i, X, i) + \sum_k \mu_k s_k(y_i, X, i))$$

其中，$t_j(y_{i-1}, y_i, X, i)$ 是转移函数，表示对于观察序列 X 的标注序列在 i 及 i-1 位置上标记的转移概率；$s_k(y_i, X, i)$是状态函数，表示观察序列 X 在 i 位置的状态概率（即标记概率）；λ_j 和 μ_k 分别是 t_j 和 s_k 的权重,需要从训练样本中学习出。

为了便于计算，我们需要使用这个公式的简化形式。请注意，公式中的相同特征是在每个位置定义的。相同的特征可以在每个位置相加，并且局部特征函数可以被转换成全局特征函数。这可以将条件随机字段写为权向量和特征向量，得到 CRF 的简化形式。

由于，我们可以将 y_{i-1} 同样作为参数加入到 s_k 的计算中，即：

$$s(y_i, X, i) = s(y_{i-1}, y_i, X, i)$$

于是，全局特征向量就可以写成：

$$F_j(Y, X) = \sum_{i=1}^{n} f_j(y_{i-1}, y_i, X, i)$$

每个局部特征函数表示状态特征或转移函数。

最终得到：

$$p(Y \mid X, \boldsymbol{\lambda}) = \frac{1}{Z(X)} \exp(\lambda_j \cdot F_j(Y, X))$$

其中 Z(X)为归一化因子。

2.2.2 概率计算

CRF 的条件计算指的是给定 CRF P(Y|X)，输入序列 X 和输出序列 Y，计算条件概率 $P(Y_i=y_i|x)$，$P(Y_{i-1}=y_{i-1}, Y_i=y_i|x)$以及相应的数学期望（E）的问题。在实现上，我们引入前向后向向量，递归地计算这些概率与期望，这种算法称为前向-后向算法。

对每个指标 i，定义前向向量$a_i(x)$为：

$$a_0(y|x) = \begin{cases} 1 & y = \text{start} \\ 0 & \text{否则} \end{cases}$$

递推公式为：

$$a_i^T\left(y_i|x\right) = a_{i-1}^T\left(y_{i-1}|x\right) M_i\left(y_{i-1}, y_i|x\right) \quad i = 1,2,\ldots,n+1$$

也即：

$$a_i^T(x) = a_{i-1}^T(\mathrm{x})M_i(x)$$

其中，$a_i\left(y_i\middle|x\right)$表示在位置 i 的标记是$y_i$，并且到位置 i 之前部分标记序列的概率，因为$y_i$可有 m 个取值，所以$a_i\left(y_i\middle|x\right)$是一个 m 维列向量。

同理，定义后向向量ß$_i(x)$为：

$$ß_0\left(y_{n+1}\middle|x\right) = \begin{cases} 1 & y_{n+1} = \mathrm{stop} \\ 0 & \text{否则} \end{cases}$$

递推公式简化结果：

$$ß_i(x) = M_i(x)ß_{i+1}(\mathrm{x})$$

$a_i\left(y_i\middle|x\right)$表示在位置 i 的标记为$y_i$。

由以上叙述，标记序列在位子 i 是标记y_i的条件概率和在位置 i-1 与 i 是标记y_{i-1}和y_i的条件概率：

$$\mathrm{P}\left(Y_i = y_i\middle|\mathrm{x}\right) = \frac{a_i^T\left(y_i\middle|x\right)ß_i\left(y_i\middle|x\right)}{Z(x)}$$

$$\mathrm{P}\left(Y_{i-1} = y_{i-1}, Y_i = y_i\middle|\mathrm{x}\right) = \frac{a_{i-1}^T\left(y_{i-1}\middle|x\right)M_i(y_{i-1}, y_i|x)ß_i\left(y_i\middle|x\right)}{Z(x)}$$

其中，$\mathrm{Z(x)} = a_n^T(x)\cdot[1 \quad \ldots \quad 1]^T = [1 \quad \ldots \quad 1]\cdot ß_1(x)$。

2.2.3 学习算法

由于 CRF 是一种对数线性模型，所以其学习方法主要为极大似然估计。具体的优化实现算法有改进的迭代尺度法 IIS、梯度下降法以及拟牛顿法。在本项目中，我采用的是与 CRF++一致的拟牛顿法与 L-BFGS 优化。

其算法伪代码表示如下：

输入：特征函数 f_1，f_2，…，f_n，经验分布 P(X,Y)

输出：最优参数值 w，最优模型$P_w\left(y\middle|x\right)$

①选定初始点w^0，取B_0为正定对称矩阵，置 k=0；

②计算$g_k = g(w^k)$。若$g_k = 0$，则停止计算，否则转③；

③$B_k p_k = -g_k$求出p_k。

④一维搜索：求 μ_k 使得：

$$f(w^k + \mu_k p_k) = \min_{\partial \geq 0} f(w^k + \mu p_k)$$

⑤置$w^{k+1} = w^k + \partial_k p_k$

⑥计算$g_{k+1} = g(w^{k+1})$，若$g_k = 0$则停止计算；否则，按下面的式子计算：

$$B_{k+1} = B_k + \frac{y_k y_k^T}{y_k^T \partial} - \frac{B_k \partial_k \partial_k^T B_k}{\partial_k^T B_k \partial_k}$$

其中，$y_k = g_{k+1} - g_k$，$\partial_k = w^{k+1} - w^k$。

⑦计算$k = k + 1$，转③。

2.2.4 特征选择

CRF 的特征模板包括两种，一元（unigram）特征和两元（bigram）特征，其中一元特征形式如“编号:%x[1,0]”指的是当前字的后一个字，“编号:%[-1,0]/%x[1,0]”指的是当前字之前一个字和之后一个字所构成的二元组。

一般比较常使用的一组基本特征模板的特征意义的解释如表 2 所示。

基本特征	特征意义
U00	当前字向前第二个字
U01	当前字的前一个字
U02	当前字
U03	当前字的后面一个字
U04	当前字后面第二个字
U05	当前字与前面两个字的三元组
U06	当前字与前一个字和后一个字的三元组
U07	当前字与后面两个字的三元组
U08	当前字与前面一个字
U09	当前字与后面第一个字

表 2 特征模板的特征意义

2.2.5 预测算法

条件概率 P(Y|X)和输入序列（又称观测序列）x，求条件概率最大的输出序列（即标记序列）y^*，即对输入序列进行标注，也即 CRF 的预测。在本项目中，所使用的 CRF 结果预测算法是著名的维特比（Viterbi）算法。

回顾之前得出的 CRF 公式：

$$p(Y \mid X, \boldsymbol{\lambda}) = \frac{1}{Z(X)} \exp(\lambda_j \cdot F_j(Y, X))$$

可知，CRF 的预测即求非规范化概率最大的最优路径：$\max_y(\lambda \cdot F(Y, X))$。

该路径又可以写成：$\max_y \sum_{i=1}^{n} \lambda_i \cdot F_i(y_{i-1}, y_i, x)$。

于是，维特比算法过程如下：

①首先求出首位字符的标记 j=1,2,…,m 的非规范化概率；

②由通项公式求出从第二位置到最后一个位置的各个标记 j=1,2,…,m 的非规范化概率的最大值；

③直到 i=n 时停止，求得非规范化概率最大值，和此最优路径的终点，进而回溯得出最优路径。

2.3 前处理与后处理

2.3.1 前处理

为了提高模型的准确性，本系统首先从对测试文本的规范性入手，主要包括：频繁串查找，用户词典，字符规范化几个方面。

2.3.1.1 频繁串查找

在我们人工观察基本的分词结果时，我们发现很多时候重复出现的短语可以作为词，并且很多时候，因为其属于新词，所以基础分词器没法完全正确切分，例如人名二次出现省略姓氏的分词问题，往往由于该名前后文语境，被分成其他词。于是在这个前处理过程中，在分词前先统计出词频大于某一个值的二字词（根据需要可以调整字数与词频阈值），然后在基础模型分词结束之后将该频繁串整合进去进一步提高分词准确性。

其算法流程大致如下：

①定义由一个字符串、整型偏置组成的结构体用于存储字符串；

②将句子切分成 N 字词数组（N 可以自定义，默认为 2）；

③扫描每个字，并将相同词的偏置加入到首次出现该词的那个结构体中；

④将每个结构体中的整型偏置数组大小大于某个值的词统计出来作为最终频繁串结果，返回。

2.3.1.2 字符规范化

因为用于训练的文本中的所有标点符号都是全角，所以当遇到测试文本中出现非全角符号时，分词器会因为标记不出来（通过训练得到的 CRF 分词器会将半角符号和前一个词分到一起，因为半角符号在训练集中一般只出现在英文中，而在英文中标点符号是跟前一个字在一起的）而将该符号归为与前一个词在一起，从而造成分词结果的错误。在预处理过程中，加入了字符的规范化，将所有符号转化成全角，这样基本可以将所有字符都拉到分词器可以识别的水平。

该步骤的算法流程大致如下：

①读取字符规范化词典，用 Map 存储；

②分词过程中，读入字符串，从头开始扫描遇到半角符号则立即转换为对应全角符号。

2.3.1.3 用户词典

在分词之前，或者说贯穿于分词过程，在 CRF 基本模型基础上，同时读取用户词典，在前处理过程中，词典的作用主要是用于训练模型。在上文中我已经提到，我在训练 CRF 模型时，不仅用到了 2 元特征，还用到了 1 元特征，于是即使是自己添加的单独的词，也可以加入训练。

2.3.2 后处理

2.3.2.1 模型自学习

由于 CRF 模型的训练过程相对而言比较繁琐，但是为了实现用户自定义学习规则进行学习这一功能，我又加入了感知机模型，在 CRF 模型基础上进行调整。

在机器学习中，感知机（perceptron）是二分类的线性分类模型，属于监督学习的方法。接受的输入是实例的特征向量，给出的输出则为实例的类别（正或负）。感知机会得出将输入空间划分为两类的分类超平面。感知机目的就是求出该超平面，为求得超平面导入了基于错误分类的损失函数，然后利用梯度下降法对损失函数进行优化。感知机的学习算法十分简单并且易于实现。感知机预测是用学习得到

的感知机模型对新的实例进行预测的，因此属于判别模型。感知机是神经网络和支持向量机的基础，是由 Rosenblatt 于 1957 年提出。

既然感知机是分类问题，这样就需要确定分词问题中，如何将每个字确定不同的分类，在后文的叙述中，将会提到现在使用最为普遍的标注方法，总共有四个不同的标注，在分类问题中，定义为四个分类，分别为 B、E、M、S。最终的预测算法也是著名的维特比（Viterbi）算法。在实现了感知机的基础上，在线学习使用的是结构化感知机实现的。正常情况下，如上所述，只能识别出二分类问题，但是因为本项目需要至少四个分类，于是对其进行了一定的调整，例如识别一个词是不是 B 类别的时候，就将其他三个类别看作负例，其他类别同理。

借助于感知机模型灵巧的学习性能，可以在 CRF 模型基础上提供在线学习功能。

2.3.2.2 自定义词典

虽然已经实现的分词器已经能在准确性上达到不错的效果，但是难免会遇到用户想要增加自己领域词典的需要。本项目实现的时候虽然在前处理中已经加入了用户词典的训练过程，但是在后处理中，仍然提供了一个添加自己词典的功能，大致原理是识别句子中存在于用户词典的词，然后将其融合到整个分词结果中。

关于词的存储，在这里需要介绍双数组 Trie 树。

根据文献[52]的介绍双数组 Trie（DAT）是 Trie 树的一种变形，这种数据结构即保证了普通 Trie 树速度，又提高了内存空间利用率。对于 DAT，每个节点代表自动机（DFA）的一个状态，根据变量的不同，进行状态转移，当到达结束状态或者无法转移的时候，完成查询，实际上查询过程就构成了一个确定状态有限自动机。实现时采用两个线性数组(代码中 baseNode 和 valueArray)，这两个数组拥有一致的下标即 DFA 中的每一个状态，也即 Trie 树中所说的节点，baseNode 数组用于确定状态的转移，valueArray 数组用于检验转移的正确性。因此，从状态 s，输入 c，再转移到状态 t 必须满足下面的条件：

baseNode[s] + c == t

valueArray[baseNode[s] + c] == s

对于给定的状态 s，如果有 n 个状态（字符 $c_1, c_2, ..., c_n$）的转移，需要在 baseNode 数组中找到一段空位 $t_1, t_2, ..., t_n$，使得 t_1-c_1, t_2-c_2, ..., t_n-c_n 都为 baseNode 数组中下标为 s 的值，注意此处的 $t_1, t_2, ..., t_n$ 不一定在 baseNode 数组中连续。对于转移的状态 $t_1, t_2, ..., t_n$，其作为下标时，valueArray[t_1], valueArray[t_2], ..., valueArray[t_n]的值都为状态 s。

在 DAT 的构造中，我采用静态构造法，将所有词语全部放入到内存，对词语中所有的父节点及其下的子节点分别进行排序，找出最初的父节点数目和有多少个不同的子节点数，方便对内存进行分配。这样的优点是找到放置子节点空间完全能够容纳子节点，以后不需要进行扩充，相对复杂度较低，且构建速度相对很快。缺点是以后添加词语不太灵活，每次需要重新构建。

整体处理过程大致如下：

①读取用户词典用双数组 Trie 树存储，并设置优先级（程序中默认设置为 1000，越大越高级）；

②使用正向最大匹配的思路，将词典中出现的词所在句子的偏移返回；

③由上一步获得一个自定义词集合，通过融合程序，将其与原始 CRF 分词结果进行融合，最终得到结果。

2.4 分词过程

基本的分词过程即主要围绕对 CRF 模型的解码过程，在此之前有简单的前处理和之后加入的其他的处理。

在进行完预处理，外部词典的加载之后，即可进入 CRF 模型的分词标注过程，在此先简述 CRF 模型为中心的序列标注预测过程：

既然是序列标注，那么首先任何字的标签不仅取决于它自己的参数，还取决于其前一个字的标签。但是因为首字前面并没有字，所以首字的处理稍有不同，假设第 0 个字的标签为 X，遍历 X 计算第一个字的标签，取分数最大的那一个。

接着我们需要解决计算其他每一个字的每个标签的概率问题。一个单词根据 CRF 模型提供的模板生成一系列特征函数。这些函数的输出值乘以函数的权重以获得分数。 这个分数只是“点函数”的得分，而“边函数”的得分需要加上。边函数在本分词模型中简化为 f(s,t)，其中 s 为前一个字的标签值，t 为当前字的标签值。于是该边函数就可以用一个 4*4 的矩阵描述，相当于 HMM 中的转移概率。

实现了评分函数后，从第二字起头便可应用维特比算法向序列后解码，为全部字打上 B、E、M、或 S 标签。

顺序扫描一遍标签，将 BM、ME、BE 合并，S 单独成词，即可得到初始的分词结果。

得到结果之后，再通过后处理过程将某些不规范的分词结果进行调整以及将用户自定义的词典识别结果如初步分词结果相融合。

2.5 高精度分词器性能

为了测试我们的高精度分词器，我们对网络上能够获取到的文本进行了测试，测试结果我将分项叙述。在这里需要点明的是，我们的模型采用的是 1998 年人工标注并校对的《人民日报》 文本训练而来，在测试时，因为材料受限，我们采用第一个月的《人民日报》文本进行测试。

2.5.1 测试前处理性能

对于同样的三个月的《人民日报》在未进行前处理的 CRF 模型分词下的结果如表 3 所示。

月份	准确率（P）	召回率（R）	F 值（F）
1998.01	0.98300	0.96011	0.97141

表 3 不加前处理的 CRF 分词结果

加入前处理过程之后，同样三个月的文本，分词结果如表 4 所示。

月份	准确率（P）	召回率（R）	F 值（F）
1998.01	0.98430	0.96975	0.97239

表 4 加入前处理的 CRF 分词结果

通过对比以上两个结果，我们可以很肯定的说，前处理对于原始模型分词有一定的改进效果。因为这些文本已经经过了人工校对，所以相对而言，已经比较正规（之前提到的前处理主要解决句子不规范的问题），所以虽然结果只有很小的进步，但其实相比而言已经算是巨大的成功。

2.5.2 测试后处理性能

在此，我仅选取 1998 年 1 月的《人民日报》已校对文本进行测试，加入的后处理主要为用户自定义词典，词典来自网络，分别有 2 万词、10 万词与 30 万词，同时这三个词典后者包含前者。测试结果见表 5。

词数	准确率（P）	召回率（R）	F 值（F）
2w	0.98300	0.96011	0.97141

10w	0.98473	0.96248	0.97348
30w	0.98582	0.96383	0.97470

表 5 加入不同规模用户词典的 CRF 分词结果

由上表我们可以看出在一定的情况下，用户词典对分词效率有一定的提高。由于词典也是从《人民日报》分词结果统计而来，而本 CRF 分词模型也是由《人民日报》校对好的文本训练而来，所以难免由于绝大多数规则已经被 CRF 模型学习而导致外部词典的增加对结果提升的效果不是很显著。

关于模型自学习的功能，我在这里用程序的截图来证明，在不添加规则的情况下（也不加任何用户词典、领域移植），分词结果见图 3。

图 3 不加自学习功能的分词结果

当添加了一些规则（“人们 审美的”、“然后 来”、“发 offer”、“买了 件”）后，得到的分词结果见图 4。

图 4 加入自学习功能的分词结果

由两图对比可见，自学习功能能够起到一定的效果。也再一次证明后处理过程有一定的作用。

为了进一步说明自学习功能的有效性，我进行了如下实验：将人工收集并校对的 103 条歧义切分文本全部加入到自学习模块中，对比结果见表 6。

测试集		准确率（P）	召回率（R）	F 值（F）
歧义切分	原始分词	0.82696	0.93071	0.87577
	自学习	0.95531	0.96067	0.95798

表 6 歧义切分中使用自学习与否的分词结果

通过以上结果发现在使用自学习模块后分词结果的 F 值增加了约 8.2%，更进一步说明了自学习功能的正确性。

2.5.3 整体性能

由于本模型主要使用《人民日报》分词结果进行训练而得，所以在评价其整体性能的时候，我们将第一个月的《人民日报》文本作为内部封闭测试集，而从网络中获取的维基百科分词文本（有很多不规范的地方，同时包含很多领域的信息，总

计约 9 万行）作为外部测试集，分别进行测试，得到的结果如表 7 所示。

测试集		准确率（P）	召回率（R）	F 值（F）
封闭测试	1 月新闻	0.98431	0.96075	0.97239
开放测试	维基百科	0.93584	0.91093	0.92322

表 7 开放与封闭测试集上的分词结果

由上表可见，在封闭测试集，同时基本不包含特定领域文本时，本系统的 F 值平均在 97%以上，已经能够作为高精度分词器使用，但是当面临开放测试集或者说带有特定领域文本的测试集时，本系统还有很大欠缺。

当然，实验结果也可以说明纯 CRF 模型在领域移植上还有些工作要做。

2.6 本章小结

本章主要阐述本项目主要使用的分词模型——CRF，在详细介绍了其分词原理的基础上，又简单介绍了保存词典的双数组 Trie 树以及在后处理过程中要用到的结构化感知机。之后，将涉及到的所有与一般文本分词相关的频繁串查找、字符规范化与词典等前处理过程以及模型自学习、用户词典辅助分词等后处理过程进行一般的介绍。后来又简单测试了前处理、后处理以及整体分词性能，进一步证明了该分词器的高性能，同时也暴露出其在领域移植上的不适应性。

第 3 章 面向领域移植的汉语分词系统设计与实现

3.1 领域移植的汉语分词

3.1.1 基本介绍

我们常见的汉语分词系统，虽然能取得很骄人的成绩，但是大多（甚至所有）分词器在面临特定领域文本的分词任务时，难免出现因未登录词过多而出现错误。领域移植，意味着我们的分词系统能够适应具体某一个或几个领域专业文本的分词任务。本项目着重为了解决医疗、金融以及法律领域的高精度汉语分词。

特定领域的高精度分词意义重大，主要体现在使用搜索引擎上，只有将句子（特定领域）完美切分，才能更进一步取得更好的搜索结果。特别是在以上列举的三个领域，正确的分词尤为重要。

3.1.2 主要问题

在上一章已经介绍了汉语分词主要问题之后，本章着重分析特定领域上的分词的主要难点。在此引入一个实例：“被告人宋皓犯受贿罪，判处无期徒刑，剥夺政治权利终身，并处没收个人财产人民币 10 万元；撤销贵州省六盘水市中级人民法院（2009）黔六中刑三初字第 32 号刑事判决第二项，改判上诉人宋皓受贿所得赃款赃物房屋及现金予以追缴，上缴国库。……”从这个特定领域（法律）文本中，我们可以看出，分词难点主要集中于未登录词的识别上，相比较而言在歧义切分上所要做的工作不是很多（本章着重处理未登录词）。

3.1.3 解决方案

由上一章，我们已经得到一个相对高精度的分词器，为了让它在特定领域作用更加完善，我在本章将介绍一个由特定领域词典、启发式规则以及命名实体识别模块组成的特定领域分词模块，用来在上一章分词器基础上再提高其在特定领域分词的准确度。

3.1 领域词典

由于本系统的特定领域分词用到了词典，而现有的资源中，很少看到有专业领域的词典存在，所以词典的构建便是很大的一部分工作。特定领域在本项目中，主要集中于法律、金融与医疗。

金融领域词典，通过网络爬虫爬取某些金融交易网站上的金融标签，制作而成。

医疗领域词典，通过医脉通的目录及附录中的病名提取而成。

法律领域词典，通过法律条文中的机构、特定名称的提取制作而成。

所有词典的格式都是“词 词性 优先级（越大越高级）”。

用户可以自由向词典中增加内容。

在实现原理上与用户自定义词典类似，使用双数组 Trie 树将词典读入内存，然后从最初的根节点开始匹配，匹配过程类似自动机（DFA）。

3.2 启发式规则

领域中的新词每时每刻都在生成，我们无法通过词典来覆盖所有的领域词汇，即使可以做到，那么由于词典过大，程序运行也比较慢。为了中和速度与准确率的问题，我在引入领域词典的基础上增加了两种规则匹配的方法，针对某些比较规则的词（例如带“《》”的书名）直接采用正则表达式进行匹配。而某些变化多样但是有规则可寻的词则采用一个自动机识别未登录的领域词汇的功能。

正则表达式的规则比较简单，在此不再赘述。

能使用基于自动机的规则匹配完美匹配出来的词必然符合一定的规律，例如在法律文本中，各级法院虽然在名字上变化很大，但是一般符合“X 市 X 级 XX 法院”这样的句式，于是为了通过一个规则就将这种类型的“词”识别出来，我的规则可能需要定义为“ns+a+law_ner”，其中 law_ner 为法律词典中存在的命名实体。

3.3 感知机辅助分词

在进行了词典结合规则匹配进行特定领域的词的识别之后，为了进一步提高领域移植分词的精度，我又加入了感知机识别命名实体的模块。

原理即为通过结构化感知机训练出词性标注模型，然后从中提取出 ns、nr、nz 等词性标注的标签。

感知机模型同样是一种概率模型，其概率计算公式为：

$$f(x) = \int wx + b$$

其中，w 为权值向量，b 为偏置。

因为其为机器学习算法，所以不免需要先进行模型的学习过程，训练样本为一个月的《人民日报》样本（因为其只是一个小的模块，不是主要功能，所以未做过多处理）。最终结果如何将在本章最后的测试结果中统一给出。

3.4 快速移植

由于本系统将领域移植部分作为单独的组件来进行编写，所以在使用时可以按要求灵活选择，也可以只选择一个领域，在系统集成的运行界面上，我们可以很简单地实现特定领域分词，也就是实现快速移植。界面见图 5。

图 5 实现快速移植的应用界面

3.5 系统整合

本章介绍的面向特定领域新词识别与未登录词识别模块在实际使用中，需要与上一节讲到的高精度分词器配合使用。

在整合过程中，原有分词器与本节叙述模块同步进行，这样我们在程序运行结束就会拿到一个分好词的字符串列表和一个由词及其偏置组成的结构体数组，在写程序时，用一个指针从字符串列表头部开始读取，遇到结构体数组中的偏置指示的位置时，将该结构体中的字符串所涉及到的词拼接或者拆开组成一个整体。这样当指针走到字符串尾部时，该结构体数组便完全嵌入到该字符串数组中了，组成最终的分词结果。

该部分算法流程：

①读取分好词的字符串列表 tmp、特定领域分词模块找出的领域词汇（带有偏置）area 以及原始字符串长度 len；

②设置临时变量 index，从下标为 0 到下标为 tmp 的长度循环：

2.1 如果 index 等于当前领域词汇的偏置，就向结果中加入该领域词汇，指针转向下一个领域词，如果没有则 break；

2.2 如果 index 比 tmp 当前词的偏置要大，就：

2.2.1 如果 index 比 tmp 下一个词偏置还要大，就跳过当前词；

2.2.2 否则，将 tmp 下一个词 index 偏置之后的部分截取出来加入到结果中

2.3 如果 index 加上 tmp 当前词长度不大于当前 area 的偏置，将 tmp 当前词加入结果；

2.4 如果 index 加上 tmp 当前词大于 area 的偏置，将 tmp 在 area 当前词偏置之前的部分截取加入结果。

③将 tmp 中其他未读取的词加入结果中。

3.5 领域移植分词系统性能

由于关于特定领域分词校对完成的文本比较稀缺，为了测试本系统在领域移植方面的性能，我们人工从网络上抓取若干条测试语句，人工分词并校对，制作成两个特定领域的分词测试集，加入特定领域分词模块与否的分词结果见表 8。

文本	准确率（P）	召回率（R）	F 值（F）

特定领域文本 1	未加模块	0.944	0.950	0.947
	加模块	0.953	0.967	0.960
特定领域文本 2	未加模块	0.944	0.961	0.952
	加模块	0.947	0.965	0.956

表 8 加入特定领域分词模块与否的测试结果

虽然用于测试的样本较少，但是足以说明本系统在加入了特定领域文本识别模块以后，分词效果有了些许提高，进而证明了特定领域分词模块的正确性。

3.6 本章小结

本章从阐述如何构建特定格式的领域词典开始，进而说明了如何结合领域词典、规则匹配、感知机等方法进行高精度特定领域专有词汇或者未登录词的识别。并简单阐述如何将本模块嵌入到整个分词系统中。最后通过简单的实验，说明了本模块的正确性。

第 4 章 评测资源与评测方法

4.1 评测资源

本次系统评测语料主要分为三个部分，一般文本、存在切分歧义的文本以及专业领域文本。其中，一般文本基本来自网络（分词参考结果不加修改），共四种：一个月的 2000 年《人民日报》共计 118515 词、网络上获近期微博数据共 30207 词、Bakeoff 2005 中北京大学提供的测试语料共计 104371 词以及微软研究院提供的语料共计 106873 词。存在切分歧义的文本是我本人从网络上搜集并人工分词整理的文本，共计 103 行。由于不同特定领域分好词的文本很难从网上获取，我们用于评测的专业领域的文本是经过我们自己人工分词校对的文本。专业领域文本涉及金融、医疗、法律，分为两个测试集，其中测试集 1 大小为 3516 词，测试集 2 大小为 3846 词。为了测试分词系统速度，通过网络爬虫爬取有实际意义的未分词文本 1630482 行大小为 285.5MB。

4.2 评测方法

按照 Bakeoff 比赛提供的计算 P、R、F 的方法对比测试我们的系统与其他主流分词系统，主要分为三个部分测试：一般文本的 P、R、F，切分歧义文本的 P、R、F 以及专业领域文本的 P、R、F。速度的评价指标设置为“词/秒”与“MB/秒”

4.3 本系统测试结果

按照以上的评测方法，在以上评测资源上，运用我们的分词程序得出的分词结果见表 9。

语料		P	R	F
一般文本	《人民日报》	0.956	0.942	0.949
	微博	0.951	0.925	0.938
	PKU	0.955	0.931	0.943
	MS	0.947	0.971	0.959
切分歧义		0.827	0.931	0.876

特定领域文本 1	0.953	0.967	0.960
特定领域文本 2	0.947	0.965	0.956

表 9 本系统测试结果

4.4 对比测试结果

本小节中，主要取大家比较常用的、分词性能相对较好的几种开源分词器（结巴分词、Stanford NLP、THULAC、LTP）进行对比，本人在编写这几个分词器测试程序时，使用的都运用的是在写这个文档时最新的 Python 库进行编写的程序，当然，所涉及的模型，完全是这些分词器自带的模型，本人没有额外增加自定义词典、更好的模型等，对比结果见表 10。

语料		分词器	P	R	F
一般文本	人民日报	结巴分词	0.864	0.809	0.836
		Stanford NLP	0.949	0.954	0.951
		THULAC	0.821	0.905	0.861
		LTP	0.983	0.983	0.983
	微博	结巴分词	0.847	0.781	0.813
		Stanford NLP	0.947	0.938	0.943
		THULAC	0.932	0.967	0.949
		LTP	0.958	0.943	0.951
	PKU	结巴分词	0.850	0.784	0.816
		Stanford NLP	0.951	0.945	0.948
		THULAC	0.944	0.908	0.926
		LTP	0.960	0.946	0.953
	MS	结巴分词	0.814	0.809	0.811
		Stanford NLP	0.843	0.882	0.862
		THULAC	0.877	0.899	0.888
		LTP	0.868	0.899	0.883
切分歧义		结巴分词	0.727	0.770	0.748
		Stanford NLP	0.657	0.747	0.699

	THULAC	0.661	0.805	0.726
	LTP	0.683	0.770	.0724
特定领域文本 1	结巴分词	0.796	0.814	0.805
	Stanford NLP	0.862	0.916	0.888
	THULAC	0.717	0.848	0.777
	LTP	0.908	0.937	0.923
特定领域文本 2	结巴分词	0.841	0.825	0.833
	Stanford NLP	0.852	0.910	0.880
	THULAC	0.737	0.858	0.793
	LTP	0.913	0.950	0.931

表 10 其他分词系统测试结果

4.5 速度测试[2]

在不加领域移植分词器模块测试时，共用时 152 秒，分出词数为 5501754 词，于是我们可以得出此时的分词速度为：361972 词/秒；1.7MB/秒。

加入领域移植模块（在此将所有模块全部加上）后分词速度测试，共用时 283 秒，分出词数为 55001767 词，于是此时的分词速度：194353 词/秒；1MB/秒。

4.6 结果分析

从自身结果来看，本系统在一般文本上的准确率一般能达到 95%以上，当然在面向歧义切分的文本时，准确率会有些许下降。在面对特定领域文本时，不出意料地准确率相对较高，比其他分词器在特定领域文本分词上有很大进步。当然，之所以会出现特定领域文本上结果比一般文本结果还要好的情况，一方面因为测试文本相对较小，另一方面，该特定领域文本经过我们人工校对，相比较从网上获得的其他一般文本，用语更加规范，分词结果更加准确。

[2] 测试平台为 8GB 内存、2GHz Intel Core i7 处理器的 MacBook Pro

结　论

本文叙述了一种面向领域快速移植的高精度分词系统的实现，主要包括基于CRF 的高精度分词系统以及一个用于特定领域快速移植的分词模块。由于特定领域的不确定性，在传统分词器遇到不同领域文本时，难免会遇到因新词存在、未登录词太多而出现错误。本文着眼于对在一般文本分词上取得喜人成果的 CRF 模型基础上通过增加特定的处理，来提高其在一般文本分词上的精度以及在面向不同特定领域文本时的快速移植。

首先，在提高原有 CRF 模型分词准确率上，我增加了诸如频繁串统计、字符正规化与加入词典训练等前处理过程以及通过感知机实现模型自学习与加入领域词典等后处理过程。在没有任何前处理与后处理过程时，分词准确率基本能达到95~96%，而加入这几个前处理之后，一般可以达到 97%，最终完整的分词器在一般文本上大概可以达到 97%以上的准确率（见上一节叙述）。之所以加入频繁串统计，是因为在分词过程中，我们发现很多时候同一个短语出现一定次数之后，就可以看做一个词，例如人名（不带姓）虽然在传统意义上不算是一个词，但是因为其必然在一个文章里出现多次，所以可以将其作为一个词，单独分割出来。之所以加入字符正规化，是因为我们发现当中文中出现半角符号时，一般来讲 CRF 模型分词时会将半角符号与前一个词放到一起，而正常来讲，我们需要将词与标点分开。传统 CRF 模型由于训练周期比较长，一般不方便进行模型自学习过程，但是借助于感知机灵巧的训练过程，可以实现这一功能。加入用户词典，算是比较简单和常见的操作了，但是在某种意义上，其确实有一定的作用，对分词准确性有一定的提高。在后一部分的特定领域分词模块的书写中同样使用到了这样的功能。

然后，我们介绍了基于感知机以及其他命名实体识别方法的特定领域快速移植分词模块。在特定领域上，主要通过领域词典、规则匹配以及感知机模型来识别该领域的未登录词，用于调整原有分词系统。从结果来看，该模块效果很好。

最后，我们展示了在一般文本、歧义切分文本以及特定领域文本等共计六个文本的测试结果，可见我们的系统在与其他开源分词器的对比中，效果显著。

关于未来工作方向，从测试结果来看，我们的分词器召回率要明显小于准确率，说明我们分出来的词要比参考结果的词要多，即我们分词器的分词粒度过小。接下来我会在加入其它模块将分散的词“合并”，同时由于个人电脑限制，CRF 模型训练语料并不够，今后会增加训练文本，训练出更好的模型。在特定领域方面，领域词典、启发式规则这些都需要不断地收集，在本项目中，主要考虑了未登录词的处

理，而分词问题方面，还有个很重要的部分——歧义切分，在本项目中没有重点解决，在接下来的工作中，将着手向这个方向努力。

参考文献

[1] 马晏. 基于评价的汉语自动分词系统的研究与实现[D]. 清华大学, 1991.

[2] 张国兵, 李淼. 一种基于局部歧义词网格的快速分词算法[J]. 计算机工程与应用, 2008, 44(12):175-177.

[3] 石佳, 蔡皖东. 基于 N 元语法的汉语自动分词系统研究[J]. 微电子学与计算机, 2009, 26(7):98-101.

[4] 韩莹, 王茂发, 陈新房,等. 汉语自动分词词典新机制—词值哈希机制[J]. 计算机系统应用, 2013, 22(2):233-235.

[5] 蒋才智, 王浩. 基于 memcached 的动态四字双向词典机制[J]. 计算机应用研究, 2011, 28(1):152-154.

[6] 刘超, 王卫东. 基于双哈希词典机制汉语分词的研究[J]. 信息技术, 2016, 40(11).

[7] 刘挺, 吴岩, 王开铸. 串频统计和词形匹配相结合的汉语自动分词系统[J]. 中文信息学报, 1998, 12(1):17-25.

[8] 唐涛. 面向特定领域的汉语分词技术的研究[D]. 沈阳航空航天大学, 2012.

[9] 卢志茂, 刘挺, 郎君,等. 神经网络和贝叶斯网络在汉语词义消歧上的对比研究[J]. 高技术通讯, 2004, 14(8):15-19.

[10] 廖先桃, 于海滨, 秦兵,等. HMM 与自动规则提取相结合的中文命名实体识别[C]// 全国学生计算语言学研讨会. 2004.

[11] 程志刚. 基于规则和条件随机场的中文命名实体识别方法研究[D]. 华中师范大学, 2015.

[12] 祝继锋. 基于 SVM 和 HMM 算法的中文机构名称识别[D]. 吉林大学, 2017.

[13] ZHUORAN WANG, TING LIU. Chinese Unknown Word Identification Based on Local Bigram Model[J]. International Journal of Computer Processing of Oriental Languages, 2012, 1(3):185-196.

[14] 原媛, 彭建华, 张汝云. 基于统计的汉语词义消歧研究[J]. 信息工程大学学报, 2007, 8(4):501-504.

[15] 肖建涛. 基于最大熵原理的汉语词义消歧与标注语言模型研究[D]. 北京机械工业学院 北京信息科技大学, 2007.

[16] 张旭. 一个基于词典与统计的汉语分词算法[D]. 电子科技大学, 2007.

[17] 佟德琴. 基于字词联合解码的汉语分词研究[D]. 大连理工大学, 2011.

[18] 赵海, 揭春雨, 宋彦. 基于字依存树的中文词法-句法一体化分析[C]// 中国计算机语言学研究前沿进展. 2009.

[19] 赵海, 揭春雨. 基于有效子串标注的汉语分词[J]. 中文信息学报, 2007, 21(5):8-13.

[20] Nianwen Xue. Chinese Word Segmentation as Character Tagging. Computational Linguistics and Chinese Language Processing, 8(1), 2003, pp. 29–48.

[21] Hwee Tou Ng and Jin Kiat Low. Chinese part-of-speech tagging: One-at-a-time or all-at-once? word-based or character-based? In Conference on Empirical Methods in Natural Language

Processing, 2004, pp. 277–284.

[22] Jin Kiat Low, Hwee Tou Ng, and Wenyuan Guo. A maximum entropy approach to Chinese word segmentation. In Proceedings of the SIGHAN Workshop on Chinese Language Processing, 2005, pp. 448–455.

[23] Huihsin Tseng, Pichuan Chang, Galen Andrew, Daniel Jurafsky, and Christopher Manning. A conditional random field word segmenter for SIGHAN bakeoff 2005. In Proceedings of the SIGHAN workshop on Chinese language Processing, vol. 171, 2005.

[24] Fuchun Peng, Fangfang Feng, and Andrew McCallum. Chinese segmentation and new word detection using conditional random fields. In Proceedings of the international conference on Computational Linguistics, 2004, pp. 562–569.

[25] Galen Andrew. A hybrid Markov/semi-Markov conditional random field for sequence segmentation. In Proceedings of the Conference on Empirical Methods in Natural Language Processing, 2006, pp. 465– 472.

[26] Ruiqiang Zhang, Genichiro Kikui, and Eiichiro Sumita. Subword-based tagging for confidence-dependent Chinese word segmentation. In Proceedings of the Annual Meeting of the Association for Computational Linguistics and the international conference on Computational Linguistics, 2006, pp. 961–968.

[27] Hai Zhao and Chunyu Kit. Integrating Unsupervised and Supervised Word Segmentation: the Role of Goodness Measures. Information Sciences, 181(1), 2011, pp. 163–183.

[28] Xiaoqing Zheng, Hanyang Chen, and Tianyu Xu. Deep learning for Chinese word segmentation and POS tagging. In Proceedings of the Conference on Empirical Methods in Natural Language Processing, 2013, pp.647–657.

[29] Wenzhe Pei, Tao Ge, and Baobao Chang. Max-margin tensor neural network for Chinese word segmentation. In Proceedings of the Annual Meeting of the Association for Computational Linguistics, 2014, pp. 293–303.

[30] Xinchi Chen, Xipeng Qiu, Chenxi Zhu, and Xuanjing Huang. Gated recursive neural network for Chinese word segmentation. In Proceedings of the Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing, 2015a, pp. 1744–1753.

[31] Jingjing Xu and Xu Sun. Dependency-based gated recursive neural network for Chinese word segmentation. In Proceedings of the Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers), 2016, pp. 567–572.

[32] Meishan Zhang, Yue Zhang, and Guohong Fu. Transition-based neural word segmentation. In Proceedings of the Annual Meeting of the Association for Computational Linguistics, 2016, pp. 421–431.

[33] Deng Cai, Hai Zhao, Zhisong Zhang, Yuan Xin, Yongjian Wu, and Feiyue Huang. Fast and accurate neural word segmentation for Chinese. In Proceedings of the Annual Meeting of the Association for Computational Linguistics, 2017.

[34] 宋彦, 蔡东风, 张桂平,等. 一种基于字词联合解码的汉语分词方法[J]. 软件学报, 2009, 20(9):2366-2375.

[35] 李寿山, 黄居仁. 基于词边界分类的汉语分词方法[J]. 中文信息学报, 2010, 24(1):3-7.

[36] Wang K, Su K Y, Su K Y. A character-based joint model for Chinese word segmentation[C]//

International Conference on Computational Linguistics. Association for Computational Linguistics, 2010:1173-1181.

[37] 王娟, 曹庆花, 黄精籼,等. 基于受限领域的汉语分词系统[J]. 信息系统工程, 2011(11):106-106.

[38] 张少阳. 领域自适应汉语分词系统的研究与实现[D]. 沈阳航空航天大学, 2017.

[39] 许华婷, 张玉洁, 杨晓晖,等. 基于 Active Learning 的汉语分词领域自适应[J]. 中文信息学报, 2015, 29(5):55-62.

[40] 苏晨, 张玉洁, 郭振,等. 适用于特定领域机器翻译的汉语分词方法[J]. 中文信息学报, 2013, 27(5):184-190.

[41] 张梅山, 邓知龙, 车万翔,等. 统计与词典相结合的领域自适应汉语分词[J]. 中文信息学报, 2012, 26(2):8-12.

[42] 朱艳辉, 刘璟, 徐叶强,等. 基于条件随机场的中文领域分词研究[J]. 计算机工程与应用, 2016, 52(15):97-100.

[43] 韩冬煦, 常宝宝. 汉语分词模型的领域适应性方法[J]. 计算机学报, 2015, 38(2):272-281.

[44] 李明. 针对特定领域的中文新词发现技术研究[D]. 南京航空航天大学, 2012.

[45] 王文荣, 乔晓东, 朱礼军. 针对特定领域的新词发现和新技术发现[J]. 现代图书情报技术, 2008, 24(2):35-40.

[46] 梁南元. 书面汉语自动分词系统--CDWS[J]. 中文信息学报, 1987, 1(2):46-54.

[47] Chen K J, Liu S H. Word identification for Mandarin Chinese sentences[J]. Proc Coling, 1992:101-107.

[48] Xue N, Converse S P. Combining Classifiers for Chinese Word Segmentation[J]. First Sighan Workshop Attached with Coling, 2002:57--63.

[49] 黄昌宁, 赵海. 中文分词十年回顾[J]. 中文信息学报, 2007, 21(3):8-19.

[50] 李正华等，中文信息处理发展报告(2016). 中国中文信息学会. 2016.

[51] Lample G, Ballesteros M, Subramanian S, et al. Neural Architectures for Named Entity Recognition[J]. 2016:260-270.

[52] 赵欢, 朱红权. 基于双数组 Trie 树中文分词研究[J]. 湖南大学学报:自然科学版, 2009, 36(5):77-80.

哈尔滨工业大学本科毕业设计（论文）原创性声明

本人郑重声明：在哈尔滨工业大学攻读学士学位期间，所提交的毕业设计（论文）《面向领域快速移植的高精度汉语分词系统研究》，是本人在导师指导下独立进行研究工作所取得的成果。对本文的研究工作做出重要贡献的个人和集体，均已在文中以明确方式注明，其它未注明部分不包含他人已发表或撰写过的研究成果，不存在购买、由他人代写、剽窃和伪造数据等作假行为。

本人愿为此声明承担法律责任。

作者签名：　　日期：　　年　　月　　日

致　谢

衷心感谢导师赵铁军教授对本人的精心指导。从本毕业设计的选题到实施，从开题到结题，我的导师都是在背后最支持我的人。每次遇到一丁点困难，他都会很热心地帮我想办法，所有与我的毕设相关的资源，他都能帮我找到并准备好。在整个毕业设计过程中，赵老师给予了我极大的支持和鼓励，他的言传身教将使我终生受益。

同时感谢本项目参考的所有开源项目及其作者们：HanLP、Ansj 中文分词、CRF++等。

再次感谢赵铁军教授，以及实验室全体老师和同窗们的热情帮助和支持！