硕 士 学 位 论 文

基于循环神经网络的中文人名识别的研究

The Research on Chinese Personal Name Recognition Based on Recurrent Neural Networks

作 者 姓 名：徐新峰

学科、专业：计算机应用技术

学　　号：21309173

指 导 教 师：黄德根　教授

完 成 日 期：2016 年 6 月 7 日

大连理工大学

Dalian University of Technology

大连理工大学学位论文独创性声明

作者郑重声明：所呈交的学位论文，是本人在导师的指导下进行研究工作所取得的成果。尽我所知，除文中已经注明引用内容和致谢的地方外，本论文不包含其他个人或集体已经发表的研究成果，也不包含其他已申请学位或其他用途使用过的成果。与我一同工作的同志对本研究所做的贡献均已在论文中做了明确的说明并表示了谢意。

若有不实之处，本人愿意承担相关法律责任。

学位论文题目：基于循环神经网络的中文人名识别的研究

作 者 签 名：徐新峰　　日期：2016 年 6 月 12 日

摘　要

中文人名识别任务是中文信息处理领域中的基础任务，其性能的好坏将直接影响到其他任务的性能。中文人名的随意性使其在未登录词中占有较大的比重，解决未登录词识别问题首先要解决人名识别问题。因此，解决中文人名识别问题具有重要的意义。

现有基于统计的中文人名识别方法存在特征选取复杂和人工干预等问题，针对这些问题，本文提出了一种基于循环神经网络(Recurrent Neural Networks)的中文人名识别方法，该方法仅采用词向量作为模型的特征且无需人工干预，有效降低了特征选取的复杂性和人工干预对实验造成的影响。此外，词向量可以通过大量未标注的中文数据训练获得，然后将蕴含丰富语义信息的词向量作为循环神经网络模型的输入，可以使模型学习到更多的信息，提升模型的性能。

本文将模型分为两个阶段：模型构建阶段和后处理阶段。

在模型构建阶段，我们将重点放在词向量的优化策略上。针对词向量的优化问题，本文提出了三种策略：

（1）将 word2vec 训练得到的词向量替换循环神经网络模型的随机初始词向量

（2）对词向量训练语料进行数词泛化操作

（3）改进 word2vec 模型，将特征信息融入词向量

实验结果表明，通过词向量的优化操作，中文人名识别模型的 F 值提高了 2.23%。

在后处理阶段，通过上下文规则对候选人名进行过滤；采用基于篇章的全局扩散操作召回在某一位置由于信息不足识别不出而在其他位置能够被识别的人名；使用基于篇章的局部扩散操作识别篇章信息中有名无姓或者有姓无名的人名。实验结果表明，通过规则过滤和扩散操作，中文人名识别模型的 F 值提高了 4.74%。

关键词：中文人名识别；词向量；循环神经网络；扩散操作

The Research on Chinese Personal Name Recognition Based on Recurrent Neural Networks

Abstract

The task of Chinese personal names recognition is fundamental in the Chinese information processing, whose performance will directly affect the other tasks. Chinese personal names account for a large proportion of the unknown word because of randomness and only if we solve the problem of names recognition firstly, can we solve the problem of unknown words recognition. Therefore, It is significant to solve the problem of Chinese name personal names.

The existing Chinese personal names recognition based on statistical methods has the problem of the high complexity of features selection and the participation of manual. In order to solve the problems, the paper proposes a method of Chinese personal names recognition based on recurrent neural network (Recurrent Neural Networks), which only uses word embedding as the feature to reduces the complexity of features selection and the impact on experimental results due to manual intervention. In addition the word embedding which is trained by unannotated Chinese data contains rich semantic information. The model will learn more information when the word embedding is the input of the model.

The model has two stages: the construction of the model and the post processing.

In the stage of the construction of the model, we focus on the optimization strategy of the word embedding and propose three strategies.

(1) Replace the random initial word embedding produced by RNN with the word embedding trained by the word2vec.

(2) Unify numeral representation on the corpus which be used to train the word embedding through numerals generalization operation.

(3) Integrate feature information into the word embedding through modifying the code of the word2vec.

Experimental results show that the results of personal names recognition increase 2.23% on F-score by optimizing word embedding.

In the stage of the post processing, we filer candidate names which are identified by Chinese personal names recognition model through using rules set to improve the precision. Besides, some names which are identified in the other position cannot be recognized due to insufficient information, so we recall unrecognized personal names according to globe diffusion operation based on chapter. We use local diffusion operations based on chapter to recall unrecognized personal names since the article use the part of the last name or the name as the name. The results show that the results of personal names recognition increase 4.74% on F-score by rule filtration and diffusion operations.

Key Words: Chinese Personal Name Recognition; Word Embedding; Recurrent Neural Network; Diffusion Operations

目　录

1 绪论

1.1 研究背景及意义

随着互联网的普及和快速发展，新数据急剧增长。大量无用虚假信息充斥在生活中，造成了信息的严重污染与浪费，增加了我们对有用信息获取的困难。此外，在海量的数据面前，我们无法通过人工处理提取信息，这时我们就需要借助自然语言处理的研究成果对数据进行分析、建模、处理，从杂乱无章的海量数据中抽取出对我们有用的信息。

自然语言处理作为人工智能领域的一个重要方向，主要研究人与计算机之间使用自然语言进行有效通信的各种理论和方法。由于自然语言文本的各个组成层次上都存在歧义性和多义性，使得实现人机间自然语言的通信是十分困难的。中文文本是由字、词、词组、句子、段、章、篇等不同粒度的信息依次构成的，不同粒度的信息在没有上下文的前提下往往表示不同的含义，因此在细粒度信息组合成更大粒度信息时都会产生多义与歧义现象。尽管当前自然语言处理中面临着很多困难，但是随着技术发展和理论提升，自然语言处理领域也在快速的发展，并取得了显著的效果。

在中文信息处理中，字不能表达具体的含义，一般将词作为最小的处理单元。而在中文文本，词并没有明显的边界特征，且由字构成词存在多种切分可能，因此中文分词在中文信息处理中成为不可或缺的工作。并且中文分词作为中文信息处理的基础性工作，其好坏将会直接影响到后续工作的性能。而在中文分词中，歧义切分问题一直是中文分词的难点，无论是基于词典分词方法还是基于统计分词方法在面对低频词问题上都没有很好的解决方案，而人名用字相对比较随意，在低频词中有较大的比重。此外，由于中文人名的随意性，导致其很难被收集到词典中，成为未登录词的一部分。未登录词的识别问题同样是中文信息处理领域的基础研究任务，且识别结果不是很理想。因此中文人名识别任务，在自然语言处理领域占据着重要的地位。

综上，中文人名识别任务是自然语言处理的基础性工作之一，有效的识别中文人名将会提升自然语言处理领域中的其他任务的性能。此外，中文人名识别在人名消歧[1]、信息抽取[2]、机器翻译[3]、文本分类[4]等众多自然语言处理领域也发挥着重要的作用。

1.2 相关研究

1.2.1 中文分词

在中文文本中，字是自然语言的最小单位，但单独的字不能表达出具体的实际意义，一般将词作为自然语言的基本单位。与英语等西方语言不同，在中文文本中词与词之间

没有明显的分词界限，从而，在进行中文文本处理时，首要处理的便是分词问题。而在英语等西方语言文本中，词与词之间存在天然的分隔符，通过空格可以很好的对词进行区分。目前中文信息处理的大部分任务都是先对文本进行分词处理，在分词的基础上进行后续操作，可见中文分词在中文信息处理领域具有重要作用。

在过去的几十年中，经过中文信息处理的不断发展以及分词需求的驱动，大量的学者投身于中文自动分词的任务中。经过长期的努力，产生了很多优异的成果。其主要的成果可以分为三类，基于词典的分词方法[5,6]、基于统计的分词方法[7-9]和基于理解的分词方法[10-13]。

（1）词典分词法

基于词典的分词方法需要构建分词词典，分词词典的质量对分词效果具有直接影响。其主要思想为：对分词字符串按照一定的匹配策略，将分词候选字符串与分词词典进行匹配，如果分词词典中存在完全一致的字串，则匹配成功，并进行下一次切分操作，直至分词完成；如果在分词词典中匹配失败，则根据匹配策略调整分词候选字符串进行下一次匹配操作。

文献[5]主要针对分词中的歧义切分问题提出了最长次长匹配算法，通过统计语料中的切分歧义发现，歧义切分字段主要出现在最长切分路径和次长切分路径中，根据此规律提出了最长次长匹配算法。该方法提高了处理歧义切分问题的能力。

词典分词法依赖于分词词典，实现简单，但对于歧义切分效果不好，并且不能对未登录词进行有效的处理。

（2）统计分词法

在中文文本中，汉字与汉字相邻共现的概率可以很好的反映　词的可信度，字与字之间共现的概率越大说明汉字之间的紧密程度越高，因此可以通过统计语料中的词频信息，根据词频信息进行共现概率、最大熵、互信息等统计量的计算，并通过设定阈值对分词结果进行筛选。

文献[7]提出了一种基于词频统计的中文分词方法，并分别使用互信息，N 元统计模型和 t-测试进行了对比分析。通过分析发现互信息一般反映的是字与字之间的静态结合，N 元统计模型主要根据前 n-1 个已知字来预测下一个字的可能，而 t 测试可以较好的反映字与字之间的动态信息。统计分词法不需要切分词典，具有很好的稳定性和跨领域能力，但是需要大量的标注文本训练模型。此外，基于统计模型的算法复杂，并且由于训练语料有限，不能很好的涵盖自然语言可能出现的各种情况。

（3）理解分词法

基于理解的分词方法主要通过神经网络模型对标注语料进行训练，从而得到分词模型。在模型的训练过程中，神经网络可以有效的学习到语法、语义信息，并将学习到的信息进行保存，经过不断的训练，使各个参数达到收敛状态。然后使用训练好的模型对未标注语料进行分词。

文献[10]提出了一种基于门递归神经网络（Gated Recursive Neural Network）的分词方法，该方法利用了复位门和更新门合并上下文的复杂组合，并使用了有监督的逐层训练方法避免扩散梯度问题，在分词效果上得到了显著的提高。

经过长时间的发展与研讨，中文自动分词取得了突破性的进展，随着研究的深入，新的问题也随之产生，出现了一些难以攻克的问题。中文分词的难点归纳为如下两点：

（1）切分歧义

歧义切分是指在进行字符串切分时，存在多种切分情况。切分歧义主要有两种：交叉型歧义和组合型歧义，且交叉型歧义在歧义切分中占有较大的比重。

① 交叉型歧义：对于字符串 ABC，其中 A、B、C 代表三个字，且 A、AB、BC、C 都可成词，那么字符串 ABC 可以切分为 A/BC 或者 AB/C，系统只有根据具体的上下文信息才能对其进行正确的切分。例如字符串“几个人”，即可以切分为“几个/人”又可以切分为“几/个人”。

② 组合型歧义：对于字符串 AB，其中 A、B、C 代表三个字，且 A、B、AB 都可成词，那么对于字符串 AB 可以切分为 AB/或者切分为 A/B。比如字符串“个人”，可以切分为“个人/”或者“个/人/”。

这两种切分歧义为最常见的切分歧义类型，且占的比重较大。由于切分歧义对分词的影响较大，因此一直是中文自动分词重点处理的问题之一。

（2）未登录词

未登录词是指没有被收录在分词词典中但必须切分出来的词，包括各类专有名词，例如人名、地名、机构名、缩写词、新增词汇、网络词汇等等。汉语是一个开放集合，专有名词组词具有随意性，另外随着互联网的快速发展，网络用语和新词大量涌现，单单依靠字典收录所有的词是不可能的。并且未登录词在分词系统中造成的切分错误要远远高于切分歧义造成的错误。此外，由未登录词产生的切分错误有可能发生扩散，波及周围词的切分。例如：

正确切分：彭楚政/事迹/报告团

错误切分：彭楚/政事/迹/报告团

其中人名用字“政”与下文“事”成词，原本“事迹”是一个正确的切分，然而由于人名“彭楚政”是一个未登录词，未登录词尾词与下文成词，从而导致“事迹”也被

切分错误。未登录词引起切分错误，并将错误放大，从而导致后续工作也发生混乱。因此，未登录词识别问题也是中文自动分词处理的重中之重。

1.2.2 人名与自动分词

文献[14]指出中文分词在过去的十年有了快速的发展，并取得了可喜可贺的成果，但仍然存在四大难点，分别为（1）词是否有清晰的界定；（2）分词与理解孰先孰后；（3）分词歧义消解；（4）未登录词识别。CIPS_SIGHAN CLP 2010[15] 中文分词评测任务的分析报告表明，参加评测的各分词系统性能的最大区别主要体现在对测试语料中未登录词的召回上，并指出未登录词的处理仍是中文分词的难点。CIPS_SIGHAN CLP 2012[16]中文分词评测任务中，任务划分出十个测试点，其中人名识别作为其中一项测试点，对分词系统进行评估，评估结果显示，在 80 个人名中，平均仅有 10 个人名切分正确。

在几次的分词任务中，都指出未登录词是中文自动分词的难点，且在中文自动分词中的切分效果不是很乐观，而人名是最常见的未登录词，占的比重较大。并且由于人名的随意性，以及随着时间的推移，人名无时无刻不在增加，将所有的人名都收录到词典中是不现实，也是不实际的，因此为了能够提高中文自动分词中对未登录词的识别效果，势必要解决中文人名识别问题。

1.2.3 中文人名自动识别

目前主要的中文人名识别方法有基于规则的方法、基于统计的方法、规则与统计相结合的方法等。

基于规则的方法主要根据中文人名的规律制定一系列的规则，通过规则集筛选人名。主要的人名规律包括人名的构成、姓氏用字、人名用字、人名的长度、人名的上下文环境信息等，研究者根据人名的规律来采用各种方法筛选并制定规则集，通过规则集来进行人名识别，在中文人名识别研究初期取得了一定的成果。

文献[17]中根据人名的组成规律，名用字是否成词规律，上下文边界词规律以及特殊姓氏在特定上下文中不能作为姓氏的规律制定了一系列规则，通过规则集对人名进行识别，在特定语料上取得了好的识别结果。文献[18]根据中文姓名的构成规律、人名常用字和上下文信息等规律建立规则集。在识别前，大量观察人工识别过程；在识别中，利用先前已经建立好的规则集对测试语料进行匹配，之后将匹配出的候选人名进行概率识别；最后利用概率模型进行筛选，得到最终结果。文献[19]提出了一种基于转换的错误驱动学习与知网相结合的中文人名识别方法，该方法将工作重点放在了上下文环境信

息以及利用错误驱动方法挑选规则上，首先通过对数据集进行上下文环境的标注，通过错误驱动和知网对标注信息制定规则集，并通过规则对人名进行筛选。

基于规则的方法实现比较简单，并且不需要人工标注语料，全面的规则集可以取得较好的识别效果，并提高系统的移植性能。但是规则与语料具有较强的耦合性，在一个语料上提出来的规则很难适应其他的语料，导致仅使用规则进行人名识别的方法往往泛化性不足，可扩展性和移植性差等问题，此外建立规则需要投入大量的人力物力，并且制定规则集的人员一般需要有一定的语言功底与语法知识，增大了提取规则集的难度。因此，实际应用中仅使用规则进行人名识别并不能取得较为理想的效果。

基于统计的方法通常是从标注语料库和人名样本库中对人名的各部分用字及其人名前后词的概率加以统计，得到这些用字被当作人名用字出现的概率，在真实文本中通过计算人名用字字串概率估值来估计候选人名是否是真实人名。该方法通常需要使用人工标注好的语料进行模型的训练，从而得到模型并通过训练好的模型对测试语料进行识别，现今比较流行的统计模型有：隐马尔可夫模型(Hidden Markov Model, HMM)、朴素贝叶斯模型(Naive Bayes)、最大熵模型(Maximum Entropy Model, ME)、支持向量机(Support Vector Machine, SVM)、条件随机场模型(Conditional Random Fields, CRFs)等模型。

文献[20]以字作为处理单位，通过知识库选择算法挑选特征，增加特征模板并使用条件随机场模型对挑选的特征进行训练，从而得到人名识别系统进行人名识别。文献[21]将统计得到的人名用字与人名上下文信息融入到朴素贝叶斯分类算法中，利用人名用字与人名上下文信息确定人名的边界，该方法充分考虑了人名前后词特征，并使用前后词特征来确定人名边界，取得较好的实验效果。文献[22]提出了一种基于支持向量机的中文人名识别方法，其按字抽取特征，构建支持向量机模型，并通过对多项式 Kernel 函数进行测试，构建最优分类超平面，从而得到人名识别模型。文献[23]主要在训练过程中加入了奖惩机制，首先使用统计的方法统计出姓用字和名用字的可信度，然后在训练时对于识别正确的姓名，分别对其姓用字和名用字以及共现词进行奖励，对于识别错误的词，分别对其姓用字和名用字以及共现词进行惩罚，最后通过比较句子的常规切分与姓名切分两种情况下的句子可信度大小，来确定最终的切分结果，并通过文本的分词效果展示人名识别的好坏，实验结果表明奖惩机制取得了令人满意的结果。文献[24]提出了基于互信息的姓名识别方法。首先对分词后的单词进行遍历，根据概率信息生成候选人名，同时计算出候选人名的概率估值，然后计算出候选人名的上下文互信息和内部互信息，最后依据概率估值来筛选候选姓名与上下文互信息和内部互信息的评价函数进行过滤，并且在抽取中还利用了动态过滤阈值来提高识别精度。

基于统计的方法可以通过对语料的训练，学习到很多有用的信息，例如：词频、词共现、人工特征、特征模板等信息。通过对语料分析，可以从语料中学习到丰富的通用信息，因此统计模型在可移植性性能上表现较好。但是统计模型一般需要人工标注的语料进行学习，而现实中人工标注的语料较少且获取比较困难。此外，统计方法对于一些出现频率不高的特殊词的识别效果并不是很好。

基于规则的方法在其移植性和扩展性上略显不足，而基于统计的方法具有较好的扩展性和可移植性，但往往有限的训练语料不足以囊括所有的情况，对于训练语料中没有出现或者出现次数少的人名，统计模型往往显得无力。因此在实际应用中，纯规则的方法和单纯使用统计的方法很少单独使用，往往要将两者结合起来使用，不仅可以解决模型的移植性问题还可以针对统计模型中的特例使用规则加以处理，两者一同使用往往可以达到事半功倍的效果。

文献[25]首先进行特征的筛选，挑选人名用字特征、上下文特征、字分类特征作为条件随机场模型的特征，编写特征模板并使用条件随机场模型对系统建模，然后使用错误驱动方法挑选规则集对人名候选进行筛选，并通过扩散操作提高人名的召回率。此外，针对中国人民、日本人名、外国音译人名分别进行了后处理操作，并取得了较好的效果。文献[26]提出了一种基于最大熵模型结合规则进行人名筛选的模型，首先从训练语料中抽取特征，并通过设定阈值对特征进行筛选，其中大于阈值的特征可以作为有效特征进行模型的训练，小于阈值的特征当作噪音特征过滤，将筛选后的特征使用最大熵模型进行训练构建模型，并通过动态词表和规则集对人名进行筛选，达到了较好的识别效果。文献[27]提出了一种在搜索日志文本中识别中文人名的方法。该方法统计搜索日志中人名内部用字，并计算人名内部用字概率，将人名内部用字概率作为条件随机场的特征进行模型的训练，然后通过计算人名可信度进行中文人名的识别，最后使用规则召回未识别的人名。

统计与规则相结合的方法可以使两种方法得到很好的补充，使系统具有更好的移植性和性能。然而，现今主流的统计方法存在特征选取复杂和人工干预等问题，通过人工的方式进行特征选取和人工编写特征模板会增加人为因素对实验结果的影响。为了解决特征选取复杂和人工干预的问题，本文使用循环神经网络模型代替现今主流的统计方法，在循环神经网络模型中，拟采用词向量作为模型的特征，以有效地降低特征选择的复杂性。

1.3 本文所做的工作

本文提出了一种基于循环神经网络(Recurrent Neural Networks)的中文人名识别方法。该方法利用蕴含丰富语义信息的大量未标注中文数据训练词向量，并将词向量融入到循环神经网络进行人名识别模型的训练，此外利用上下文规则和正向逆向匹配算法对识别结果进行修正，最后通过扩散操作将未被识别的人名召回。实验证明，该方法具有较好的性能。

本文主要工作内容如下：

（1）将人名识别问题转化为序列标注问题，并使用循环神经网络进行模型的训练。循环神经网络可以通过隐藏层对所训练的数据信息进行存储，序列标注时可以根据存储在隐藏层中丰富的数据信息进行标注。循环神经网络模型可以长期地保存并利用上下文信息，对于序列预测非常有益。

（2）为了解决训练语料不足的问题，本文通过 word2vec 对大规模语料进行训练得到词向量，并使用训练好的词向量替换模型随机初始的词向量。此外，由于数词在语料中出现的频率比较高，且数值多变不容易控制，在模型训练阶段每一个数词都具有单独词向量，增加了模型的复杂度，因此本文对语料中的数词进行泛化处理。

（3）由于词向量信息中包含最多的是上下文信息，其人工特征信息比较匮乏，因此本文对 word2vec 进行改进，将人工特征信息注入到词向量中，使词向量包含更丰富的信息，从而提升系统的性能。

（4）对模型识别的人名进行后处理操作，首先将模型识别的人名转换为字符串定为人名候选，然后使用规则集对人名候选进行筛选，最后通过基于篇章的扩散操作识别未被识别的人名。

1.4 本章小结

本章主要介绍了中文人名识别的背景及其意义，然后介绍了中文自动分词的发展，并指出了中文人名识别问题与中文自动分词的关系，接着介绍了中文人名识别任务中比较成熟的识别方法，对各类方法的优缺点进行了点评，最后对自己所做的主要工作以及论文的组织结构进行了简要介绍。

2 理论基础

2.1 词向量

2.1.1 One-hot 词表示法

要将自然语言处理的问题转换为机器学习的问题，首要处理的就是要将自然语言进行符号数字化。其中最直观、最常用的词表示法是 One-hot 表示法，这种表示方法把词表示为一个很长的向量，向量的维度是词表的大小，其中绝大多数元素为 0，只有一个维度的元素为 1，这个维度就代表了当前的词。例如，

“北京”的 One-hot 表示为：[0 0 0 0 1 0 0 0 0 0 0 0 …]

“杭州”的 One-hot 表示为：[0 0 0 0 0 1 0 0 0 0 0 0 …]

由上述例子可以看出，One-hot 表示法表示的矩阵存在稀疏问题，在一个 One-hot 表示中只有一个 1，其他维度全为 0，因此我们可以使用采用稀疏方式存储，为每一个词分配一个数字 ID，那么“北京”的 ID 为 4，“杭州”的 ID 为 5，每一个词都可以使用一个数字 ID 来代替。这种简洁的表示方法已经配合最大熵、SVM、CRF 等算法完成了自然语言处理领域中的各种主流任务。

然而 One-hot 表示法存在以下两个问题：

（1） 语义鸿沟

One-hot 表示法只是表示一个词是向量所有维度中的一维，或者说一个词对应的就是一个数字 ID，在向量中看不出词的其他特征，即便是两个近义词，在 One-hot 表示法中也不能发现任何特征表明两个词之间存在关系。

（2） 维度灾难

One-hot 表示法中向量的维度为词表的大小，换句话说词个数越多，向量的维度就会越大。向量的维度越大，向量的运算就会越困难，因此当词表过大时，将会引起维度灾难。

2.1.2 词向量表示法

在深度学习领域，一般采用分布式表示（Distributed Representation）的方法表示词，此方法表示的词一般称为词向量。使用分布式方式表示词的思想最早是 Hinton[28] 于 1986 年提出的。

词向量很好的克服了 One-hot 词表示法的不足，它可以将词表示为低维向量，向量的维度不受限于词表的大小。并且词向量将每一个词表示为空间中的一个点，空间的维

度可以自己定义，而空间中点与点的距离可以体现两个词的相似度，距离越近，表示两个词越相似。例如词“北京”与词“杭州”的词向量如下：

“北京”的词向量表示为：

[0.1115257, 0.03174963, -0.0606447, -0.2004882, 0.12543263, -0.12895702, …]

“杭州”的词向量表示为：

[0.1353249, -0.0551172, -0.0363237, -0.2077684, 0.06146560, -0.10832328, …]

计算“北京”与“杭州”的相似度为 0.764431（相似度取值为[0, 1]），表明两词在空间中的距离特别近，同时也说明两者在语义上较为相近。

词向量是语言模型训练的产物，可以将大量的未标注的普通文本作为语言模型的输入，语言模型通过统计词频、词共现、词搭配、语法、语义等信息进行精细的分析，从而学习出词向量。由于现今标注语料较为稀缺，而训练词向量只需未标注的普通文本，因此词向量的获取较为容易。目前很多的自然语言处理任务中都采用词向量表示词，且取得了很好的效果。

2.1.3 word2vec

本文词向量的训练采用的是 word2vec 工具，该工具由 Google 于 2013 年开源推出。其主要的作用就是将词映射到一个 m 维的向量中，使用向量来代表一个词，如此一来，自然语言处理的问题即可转换为对向量处理的问题。通过词向量我们可以更好地对自然语言处理任务进行建模处理。

word2vec 中主要使用了两种模型：CBOW (Continuous Bag-of-Words Model) 和 Skip-gram (Continuous Skip-gram Model) [29]。两种模型的示意图分别如图 2.1 和图 2.2 表示。

由图 2.1 和图 2.2 可知，两种模型都包含三层：输入层、映射层、输出层。$w(t)$代表当前词，$w(t-2)$、$w(t-1)$、$w(t+1)$、$w(t+2)$表示词 $w(t)$的上下文。CBOW 模型是已知当前词 $w(t)$的上下文 $w(t-2)$、$w(t-1)$、$w(t+1)$、$w(t+2)$的前提下预测词 $w(t)$，而 Skip-gram 模型恰恰相反，是在已知当前词 $w(t)$的前提下预测其上下文信息 $w(t-2)$、$w(t-1)$、$w(t+1)$、$w(t+2)$。

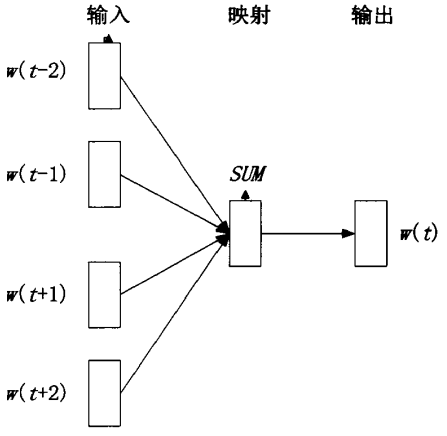

图 2.1 CBOW 模型结构

Fig. 2.1 The CBOW model architecture

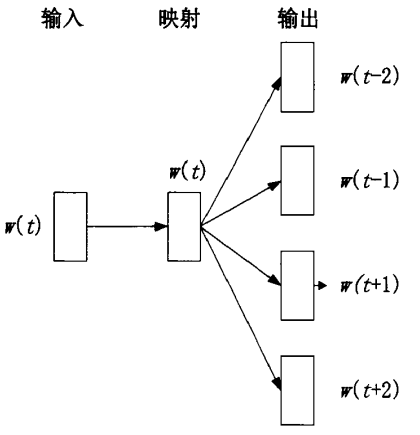

图 2.2 Skip-gram 模型结构

Fig. 2.2 The Skip-gram model architecture

以基于 Hierarchical Softmax 的 CBOW 模型和 Skip-gram 模型为例进行介绍。

（1）CBOW 模型

CBOW 模型的主要思想是已知当前词 $w(t)$的上下文的前提下预测词 $w(t)$。CBOW 模型的网络结构如图 2.3 所示。

由图 2.3 可知，CBOW 模型主要由输入层、映射层、输出层三部分组成，其中输入层输入当前词 $w(t)$的上下文 $2c$ 个词的词向量；映射层使用公式 2.1 将 $2c$ 个词的词向量进行相加，结果存入 x_w 中；输出层对应一棵霍夫曼树，其中叶子节点代表语料中的每一个词，非叶子节点代表一个二分类器，x_w 从根节点通过 k 个（k 表示从根节点到当前词叶子节点路径上的非叶子节点个数）二分类器进行分类，每经过一次分类器都会将被分到正确分支的误差累积到向量 *neule* 中，经过 k 次分类之后，x_w 信息到达当前词的叶子节点，然后将误差累积向量 *neule* 中保存的信息分别更新到 $2c$ 个词向量中。

$$X_w = \sum_{i=1}^{2c} v(context(w_i)) \tag{2.1}$$

CBOW 模型中，最关键的处理就是输出层上的操作，输出层对应一颗霍夫曼树，每一个非终端节点都可以看作是二分类问题，而二分类问题可以使用逻辑回归函数进行分类，其公式如 2.2 所示：

$$\sigma(x) = \frac{1}{1+e^{-x}} \tag{2.2}$$

逻辑回归函数的取值范围为(0,1)，阈值定为 0.5，大于 0.5 代表正例，小于 0.5 代表负例，逻辑回归函数可以很好的处理二分类问题。

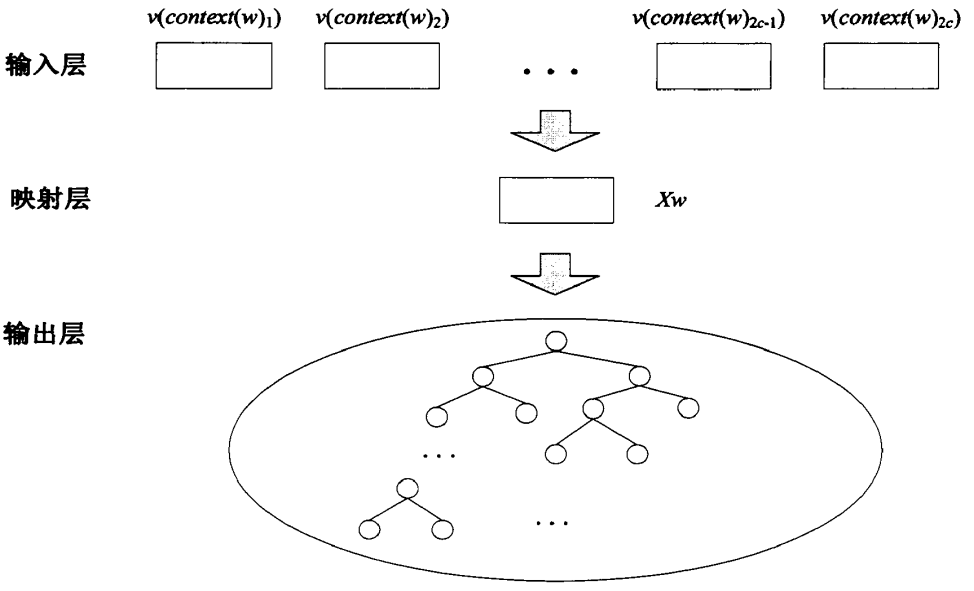

图 2.3　CBOW 网络结构

Fig. 2.3　The network structure of CBOW

对于每一个词 $w(t)$, 从根节点 root 到词 $w(t)$的叶子节点都会有一条唯一的路径 ℓ，假设路径 ℓ 中有 k 个非叶子节点，那么就需要进行 k 次二分类，如果要保证 k 次二分类问题效果最好，可以将目标函数取为最大似然估计，其公式如 2.3 所示

$$p(w|Context(w)) = \prod_{j=1}^{k} p(d_j^w \mid x_w, \theta_{j-1}^w) \tag{2.3}$$

其中 θ 代表待定参数，其值保存在非叶子节点中，只要待定参数能够确定了，代表模型训练完成。条件概率表示在向量 x_w 和待定参数为 θ 的前提下，分类为 d_j 的概率，d_j=1 代表正例，d_j=0 代表负例。

（2）Skip-gram 模型

Skip-gram 模型的主要思想为使用当前词对上下文信息进行预测。其网络结构如图 2.4 所示。

为了与 CBOW 比较，在图 2.4 中画出了映射层，其实在 Skip-gram 模型中不存在映射层。因此，Skip-gram 模型主要由输入层和输出层二层组成。其中输入层为当前词 $v(w)$

的词向量，输出层同 CBOW 模型一样对应一棵霍夫曼树。与 CBOW 的不同之处在于，在 $v(w)$从根节点经过 k 个二分类器后，将误差累积向量 *neule* 中保存的信息更新到词 w 的词向量 $v(w)$中。

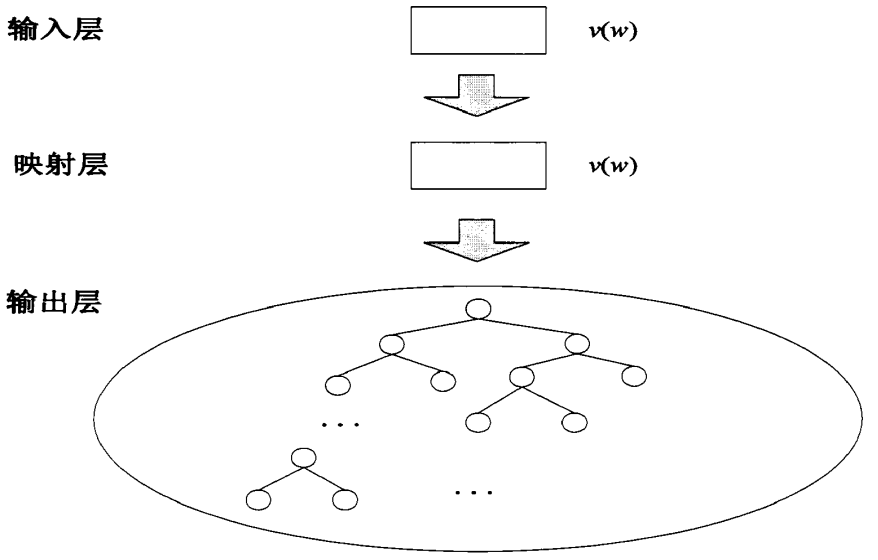

图 2.4　Skip-gram 网络结构

Fig. 2.4　The network structure of Skip-gram

Skip-gram 模型的目标函数与 CBOW 模型的目标函数不同，其目标函数如公式 2.4 所示：

$$p(context(w)\,|\,w)=\prod_{j=1}^{k}p(d_j^u\,|\,v(w),\theta_{j-1}^w) \tag{2.4}$$

其中θ代表待定参数，其值保存在非叶子节点中，只要待定参数θ确定了，其最终的模型也就训练完成。条件概率表示在向量 $v(w)$和待定参数为θ的前提下，分类为d_j^u的概率，d_j^u=1 代表正例，d_j^u=0 代表负例，这里 u 代表当前词 w 的上下文中的某一个词。由于 Skip-gram 模型是在已知当前词 w 的前提下预测上下文信息，而上下文信息中包含多个词，因此与 CBOW 模型不同的是，Skip-gram 模型的每次训练需要对 *win* 条路径进行预测（*win* 为上下文 *context*(*w*)中词的个数），每条路径代表从根节点到 *context*(*w*)中的某一个词所对应的叶子节点。只要确定预测哪一个上下文 u，路径既可以唯一确定，d_j^u也就确定。

例如上下文信息使用 $w(c\text{-}2)$、$w(c\text{-}1)$ … $w(c+1)$、$w(c+2)$表示，那么在 Skip-gram 模型中需要分别计算 $p(w(t\text{-}2)|w(t))$、$p(w(t\text{-}1)|w(t))$… $p(w(t+1)|w(t))$、$p(w(t+2)|w(t))$，并将每次的误差累积向量 *neule* 中的值更新到当前词向量 $v(w)$中。

word2vec 不仅实现了基于 Hierarchical Softmax 的 CBOW 模型和 Skip-gram 模型，还实现了基于 Negative Sampling[30]的 CBOW 模型和 Skip-gram 模型。此外，word2vec 还对低频词、高频词以及学习率的调整做了相应的处理，取得了很好的效果。

2.2 循环神经网络（Recurrent Neural Networks）

2.2.1 神经网络模型

现今神经网络模型已经在众多自然语言处理任务中取得了巨大的成功，并获得了广泛的应用。在研究神经网络模型的初期，就已经有学者将人工神经网络应用到序列预测领域，然而神经网络模型首次应用于语言处理领域是 Jeff Elman[31]所做的研究，Jeff Elman 利用神经网络将人工语法生成的词构造成句子。使用神经网络处理自然语言任务最经典的案例是 Bengio[32]等人提出的，Bengio 使用三层神经网络构建了一种基于语言的统计神经网络模型，并同 n-gram 模型、基于分类的模型做了对比。其网络模型如图 2.5 所示。

由图 2.5 可知，最下层的 $w_{t-n+1}\dots w_{t-2}$、w_{t-1}为词 w_t 的前 n-1 个词，这一层输入词的编号。然后由下到上分别为输入层、隐藏层、输出层，这三层就是神经网络模型的三层结构。输入层会根据最下层输入的词编号从词向量矩阵 C 中取出相应的词向量，词向量矩阵 $C \in R^{|V|\times m}$，其中$|V|$为词典的大小，即为语料的总词数，m 为向量的维数。并将前 n-1 个词的词向量首尾相连，组成一个$(n\text{-}1)m$ 维的向量，记为 x_w；隐藏层将输入层的输出作为输入，进行矩阵转化，并使用双曲线正切函数 *tanh* 函数作为激活函数，将结果作为隐藏层的输出并输出到输出层，其计算公式如 2.5 所示：

$$z_w = \tanh(Wx_w + p) \tag{2.5}$$

其中 W 为输入层到隐藏层的权重矩阵，p 为输入层到隐藏层的偏移量。

输出层一共有$|V|$个节点，每一个节点 y_i 代表词为 i 的未归一化 *log* 概率，其输出值 y 的计算公式为如 2.6 所示：

$$\mathrm{y}_w = Uz_w + qp(w|Context(w)) = \frac{e^{y_{w,i_w}}}{\sum_{i=1}^{|V|} e^{y_{w,i}}} \tag{2.6}$$

其中 U 为隐藏层到输出层的权重矩阵，q 为隐藏层到输出层的偏移量。

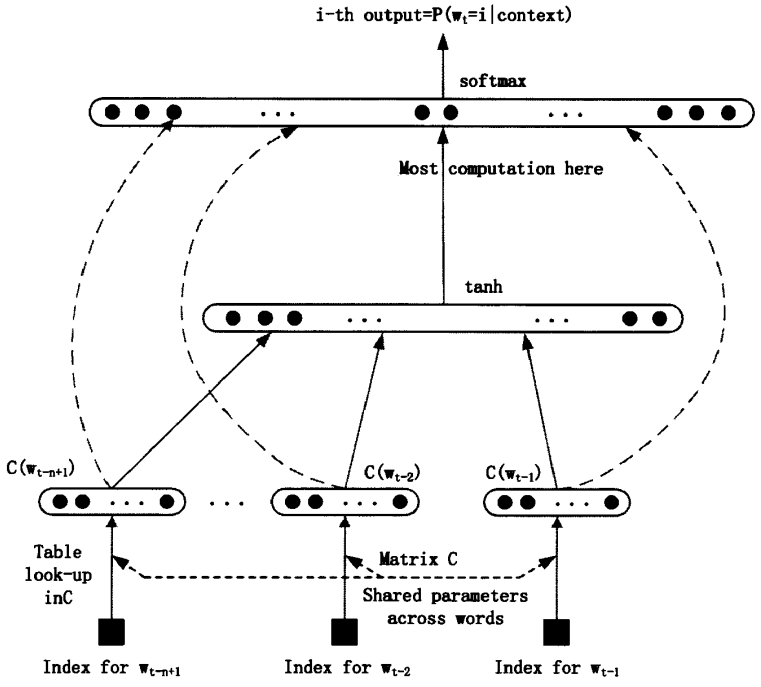

图 2.5　三层神经网络模型

Fig . 2.5　The model of three-layer neural networks

经过神经网络的训练，最终在输出层得到 $y_w=(y_{w,1},y_{w,2},\ldots,y_{w,|V|})^T$，$y_w$ 是一个长度为 $|V|$ 的向量，其分量并不能表示概率，因此需要通过 softmax 归一化处理，处理后 y_w 的分量即表示当上下文为 *context(w)* 时，下一个词恰为词词典 V 中第 i 个词的概率，其 sofmax 归一化公式如 2.7 所示：

$$p(w|Context(w))=\frac{e^{y_{w,i_w}}}{\sum_{i=1}^{|V|}e^{y_{w,i}}} \tag{2.7}$$

神经网络与 n-gram 模型相比主要有三点不同：

（1）n-gram 模型的输入都是已知的，而神经网络中输入为词向量，词向量也是一种参数，需要在模型的训练过程中不断的调整。

（2）在神经网络模型中，词语的相似性可以通过词向量来体现，例如：

例句 s1：新华社 记者 戴 浩 摄

例句 s2：新华社 记者 张 宿堂 摄

假定例句 1 在语料中出现了 10000 次，例句 2 在语料中出现 1 次。按照 n-gram 模型的操作，p(s1)远远大于 p(s2)。但是 s1 与 s2 的唯一区别在“戴 浩”与“张 宿堂”不同，这两个人名无语是在语法还是语义上都扮演着相同的角色，因此 p(s1)与 p(s2)应该相近才对。

在由神经网络训练的模型中可以得到 p(s1)与 p(s2)大致相等。因为在神经概率语言模型中，“相似”的词其对应的词向量也是相似的。此外，概率函数关于词向量是平滑的，即词向量的一个小变化对概率的影响也是一个小的变化，因此在神经网络模型中，s1 出现的次数远远高于 s2 的情况下，如要 s1 多出现一次就会增大 p(s2)的概率。

（3）神经网络模型自带平滑，其概率取值范围为(0,1)，不必像 n-gram 模型在平滑问题上多做处理。

2.2.2 循环神经网络

简单的神经网络仅仅通过特定的前几个词来预测当前词，而不能利用前几个词之外的信息，且网络模型结构中仅仅是层与层之间有关联，在同层之间没有数据交互。循环神经网络很好地解决了这个问题，它将所有的信息保存在隐藏层，并通过隐藏层提供给下一层使用，因此循环神经网络可以根据从训练数据集中学习到的丰富信息来预测当前词。循环神经网络框架如图 2.6 所示：

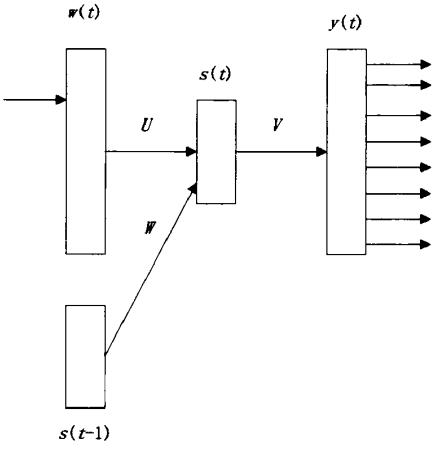

图 2.6 循环神经网络框架

Fig . 2.6 The frame of RNN

循环神经网络由输入层、隐藏层和输出层三层构成，其中 $w(t)$表示当前词的词向量，$s(t-1)$表示前一步隐藏层的输出，$y(t)$表示当前词训练后的输出，W、U 和 V 分别表示前一步隐藏层与当前隐藏层的权重矩阵、输入层与当前隐藏层的权重矩阵和当前隐藏层与输出层的权重矩阵。输出值 $y(t)$的计算公式如公式 2.8 和公式 2.9 所示：

$$s_j(t) = f\left(\sum_i w_i(t)u_{ji} + \sum_l s_l(t-1)w_{jl}\right) \tag{2.8}$$

$$y_k(t) = g\left(\sum_j s_j(t)v_{kj}\right) \tag{2.9}$$

其中，$f(z)$与 $g(z)$分别表示 sigmoid 函数和 softmax 激活函数，其中 softmax 函数主要将输出层的值表示为概率形式，$f(z)$与 $g(z)$的计算公式如公式 2.10 所示：

$$f(z) = \frac{1}{1+e^{-z}}, \qquad g(z_m) = \frac{e^{z_m}}{\sum_k e^{z_k}} \tag{2.10}$$

使用权重矩阵表示公式，将公式 2.8 和公式 2.9 分别重新表示，其表示如公式 2.11 和公式 2.12 所示：

$$s(t) = sigmoid(Uw(t) + Ws(t-1)) \tag{2.11}$$

$$y(t) = soft\max(Vs(t)) \tag{2.12}$$

在循环神经网络中，隐藏层的输入不仅仅依赖于当前步的输入层，还依赖于上一步的隐藏层，上一步的隐藏层中存储了从之前训练语料中学习到的有用信息，其有用价值与输入层相比更加的丰富，因此在模型构建时，不应该只优化对当前词的预测问题，还应该对隐藏层信息优化问题加以重视。而普通的 BP 算法并没有对前一步的隐藏层信息作出优化处理，因此使用普通的 BP 算法训练循环神经网络模型，无法对上一步隐藏层的信息进行优化。为了解决这个问题，在 BP 算法中加入时间维度，我们称之为 BPTT[33] 算法，使用 BPTT 算法即可很好的解决上述问题。BPTT 算法的网络模型示意图如图 2.7 所示。

对于训练集中的每一个示例，期望模型预测的值与真实值的差值低于某一阈值，将预测值与真实值的差记为误差，其公式如公式 2.13 所示：

$$e_o(t) = d(t) - y(t) \tag{2.13}$$

其中，$d(t)$为当前词 $w(t)$的目标向量，而 y(t)为模型对当前词 $w(t)$的预测向量，通常我们会根据 $e_o(t)$ 反向调节权重矩阵的值，目的为是 $e_o(t)$ 能够得到最小，如果 $e_o(t)$ 不在我们可以接受的范围之内，就需要进行误差的反向传播，逆向调整各个权重矩阵的值。

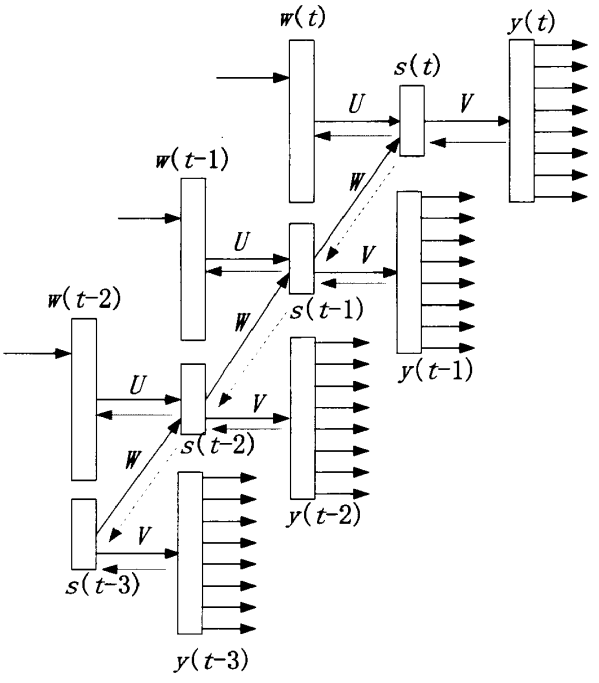

图 2.7　BPTT 算法网络模型

Fig. 2.7　The model of BPTT

其中，反向剪头代表误差的反向传播。

从图 2.7 中可以看出，误差从当前步输出层 $y(t)$反向传播到当前步的隐藏层 $s(t)$对矩阵 V 进行更新；从当前步隐藏层 $s(t)$反向传播到上一步隐藏层 $s(t\text{-}1)$对权重矩阵 W 进行更新；从当前步隐藏层 $s(t)$反向传播到当前层输入层 $s(t)$对权重矩阵 U 进行更新。权重矩阵 U、V、W 的更新公式如公式 2.14 所示：

$$\begin{aligned} v_{jk}(t+1) &= v_{jk}(t) + s_j(t)e_{ok}(t)\alpha - v_{jk}(t)\beta \\ u_{ij}(t+1) &= u_{ij}(t) + w_i(t)e_{hy}(t)\alpha - u_{ij}(t)\beta \\ w_{ij}(t+1) &= w_{ij}(t) + s_i(t-1)e_{hy}(t)\alpha - w_{ij}(t)\beta \end{aligned} \tag{2.14}$$

通过对整个数据集的训练，循环神经网络不断的调整各权重矩阵的参数，最终达到收敛状态。

循环神经网络模型已经在语言模型与文本生成领域[34-36]、机器翻译领域[37-39]和语音识别领域[40]取得了显著的成果。

2.3 本章小结

本章对词向量和循环神经网络模型进行了详细的介绍。其中在词向量小节中介绍了词向量的优势以及原理，在循环神经网络小结中对神经网络的基本思想进行了简要的概括。此外，词向量与神经网络模型相辅相成，高质量的词向量可以为神经网络模型的训练提供更加有用的信息进行学习，而好的神经网络模型又能训练出高质量的词向量，掌握了词向量和神经网络的原理，可以更好的对模型进行改进，提升系统的性能。

3 中文人名识别

中文人名主要包括汉族人名、日本人名、少数民族译名以及外国人名译名。由于人名用字的随意性，使中文人名成为最常见的未登录词，且在未登录词中占有较大的比重，因此，解决中文人名识别问题可以提高未登录词的识别性能。

命名实体识别一般包括三大类：实体类、时间类和数字类。根据 MUC-7 的定义，命名实体分为人名、地名、机构名、时间、地点、日期、货币和百分比等 7 种类型，通过识别文本的命名实体即可了解文本的主要内容，因此命名实体识别是信息处理的一项关键基础技术。而人名作为命名实体的一部分，具有重要的研究意义。

3.1 中文人名识别的难点

人名的随意性给中文人名识别带来了很大的困难，主要难点总结如下：

（1） 中文人名没有明显的边界特征

在英文文本中，人名的首字母使用大写，根据此特征就可以很好的定位人名的左边界。而在中文文本中，人名和其他的词一样，没有明显的边界特征，且人名在句首、句中和句尾均会出现。

（2） 人名用字与上下文成词

人名用字与上下文成词主要分为姓氏与上文成词、姓氏与名成词、名用字成词、名尾字与下文成词四种情况，其中名用字成词对人名识别几乎没有影响，主要介绍另外三种成词情况。

① 姓氏与上文成词

姓氏与上文成词，会在人名识别中弱化姓氏的特征，导致人名不能正确的识别，即便识别出来，被识别的人名也会存在边界问题。例如：

“李 清 白马 德 祖 摄影 报道”

其中，姓氏“马”与上文“白”成词，导致人名“马德祖”识别不出，或被识别为“白马德祖”，对人名识别带来了不便。

② 姓氏与名成词

姓氏与名成词，同样会弱化姓氏的特征，导致特征不明显，人名不能被识别的问题，但此类成词情况不会导致姓名识别的边界出现问题。例如：

“王国 栋 嘴里 答应 着”

其中，姓氏“王”与名用字“国”成词，导致“王”作为姓的特征被弱化，增加了人名识别的难度。

③ 名用字与下文成词

名用字与下文成词会导致人名识别的后边界出现问题，以至于人名识别错误。例如：

“褚 时 健在 自己 的 晚年”

“赵 孝 东家 两 个 暖棚 种 了 1 2 0 0 棵 牡丹”

其中，人名尾字“健”与下文“在”成词为“健在”，人名尾字“东”与下文“家”成词为“东家”，最终导致人名识别为“褚时健在”和“赵孝东家”，且此类错误要比①类错误多的多，在人名识别错误中占有大的比重。

（3） 中文人名的种类繁多，构成方式复杂多样。主要针对中国人名、日本人名、外国音译人名进行分析，并根据其姓氏用字个数及名用字个数进行分类。

① 中国人名

中国人名又包括汉族人名和少数民族音译人名。其中汉族人名主要由{姓氏+名字}的组成方式构成，除此之外还有有姓无名、有名无姓、笔名、绰号等特殊情况。少数民族音译人名没有固定的构成形式。中国人名的主要构成如表 3.1 所示：

表 3.1 中国人名构成

Tab. 3.1 Structure of Chinese personal name

构 成	举 例
单姓单名	广东吕钦
单姓双名	办公室主任林瑞康
双姓单名	欧阳修说
双名双姓	欧阳琦琳
单姓无名	梅的“移步不换形”论
双姓无名	欧阳说
单名无姓	娟，过来下
双名无姓	照顾小招弟
冠夫姓单名	记者彭张青
冠夫姓双名	主席范徐丽泰
笔名、绰号	冰心、巴金
少数民族人名	西藏自治区体委主任姬嘉表示

② 日本人名

日本人名与中国人名的组成结构相同，都是由{姓氏+名字}组成，不同之处在于，日本的姓氏大多为双字姓氏，单字姓氏较少，人名的长度较中国人名较长，且名用字长

度不固定，因此不再区分单名多名情况，且日本人名中一般使用姓氏表示人名。日本人名的主要构成如表 3.2 所示：

表 3.2　日本人名构成

Tab. 3.2　The form of Japanese name

构成	举例
单姓	副社长林实
双姓	日本首相桥本龙太郎
有姓无名	桥本表示相信

此外，日本姓氏可以随意地改变，可变性高，增加了日本人名的识别的难度。

③ 外国音译人名

外国音译人名没有明显的构成规律，且人名长度长短不一。例如：“普京”、“米卢蒂诺维奇”、“斯特凡诺普洛斯”等等。

（4）人名与上下文产生交叉型歧义

例如“张玉爱读小说”，可以认为“张玉”为人名，也可以认为“张玉爱”为人名，单凭这一句话不能够区分到底哪一人名切分是正确的，必须根据上下文信息做出正确的判断。

3.2　中文人名特点

本文对 2000 年《人民日报》新闻语料进行统计分析，对中文人名的特点进行总结。

3.2.1　姓氏用字规律

中国汉族人名和日本人名都是由{姓氏+名字}的形式构成，且姓氏用字比较集中，因此，姓氏在中国人名和日本人名的识别中是一个很重要的特征。

中国汉族人名姓氏规律如下：

（1）中国姓氏一般由单姓和复姓构成，且复姓较少，常用复姓有：欧阳、上官、司徒、皇甫、令狐、长孙、相里和诸葛。中国汉族人名具有明显的长度特征，一般为 2-4 个字，且 4 字人名较少。除此之外，姓氏用字比较集中。在对 82130 个中文人名统计信息中，出现单姓为 542 个，复姓 9 个，复姓出现次数仅占总人名数的 0.001%。按照出现次数从大到小进行排序，其中前 100 的高频姓氏及其所占的比例如表 3.3 所示：

表 3.3　前 100 个高频姓氏及所占比例

Tab. 3.3　The first 100 high frequency surnames and proportion

姓氏排序号	姓氏用字	所占比例
1-5	李、王、张、江、刘	31.32%
6-15	朱、陈、吴、杨、周、赵、胡、马、孙、徐	20.65%
16-65	何、邓、金、黄、钱、毛、郑、唐、罗、许、高、吕、宋、林、董、郭、丁、谢、曾、韩、温、曹、叶、成、贾、袁、田、于、尉、彭、姜、汪、傅、白、任、蒋、梁、肖、杜、迟、范、石、雷、邹、卢、冯、沈、蔡、孟、潘	29.08%
66-100	顾、陶、龚、崔、孔、程、戴、姚、熊、苏、魏、陆、秦、余、方、薛、谭、万、侯、常、夏、齐、武、钟、尹、贺、施、阎、邵、乔、洪、严、郝、曲、耿	7.94%

（2）由表 3.3 可知，即便中国姓氏很多，但是使用较为集中，其中前 15 个姓氏使用率已经占到了所有姓氏的 51.97%，超过总姓氏用字的一半。

（3）部分姓氏用字还可以作为普通用字，比如王、张、江、何等。

日本人名中，一般以地名、自然现象、宗教信仰等作为姓氏，且排名前十的姓氏占总人口的 10%，可见，日本姓氏使用同中国姓氏一样集中在少数姓氏上。

3.2.2　名字用字规律

（1）由于人名容易受到社会背景、历史事件等因素的影响，因此名字用字相对于姓氏用字比较分散，但名字用字也会集中在少数字上。统计数据表明单名用字 1095 个。按照单名用字出现的次数从高到低进行排序，前 100 的高频单名用字及其所占的比例如表 3.4 所示。

双名用字总共 13197 个，将双名用字分为双名首字用字和双名尾字用字，并对首字用字与尾字用字进行统计并排序，其中双名首字用字共 1326 个，双名尾字用字共 1455 个，其双名首字用字前 100 的高频用字及其所占比例如表 3.5 所示。双名尾字用字前 100 的高频用字及其所占的比例如表 3.6 所示。

（2）由表 2.5 和表 2.6 可知，名字用字在不同的位置上出现比重是不同的，名字内部的位置信息对人名识别也很重要。

（3）名字设计的范围非常广，有介词、连词、副词、数字等。

（4）表示死亡、疾病、邪恶含义的字，如：死、瘦、饥、苦、怪、邪等是人们所忌讳的，一般不用作名用字。

表 3.4　高频单名用字及所占比例度

Tab. 3.4　High frequency single name and proportion

单名用字序号	单名用字	所占比例
1-10	鹏、军、平、伟、仪、健、杰、锋、文、勇	24.67%
11-50	明、琳、刚、波、斌、磊、涛、干、雯、扬、强、飞、毅、华、菊、云、青、原、楠、飚、旭、峰、彤、志、俊、进、宁、辉、倪、亮、佳、岩、红、昊、跃、克、兢、英、雪、宏	26.52%
51-100	霞、凯、敏、林、政、群、静、建、钢、江、琼、坚、力、裕、宪、德、淇、洋、娜、芳、捷、晖、帆、鲁、利、岐、忠、悦、威、韬、冰、川、燕、璇、盘、洁、森、震、衍、海、丽、琪、舸、征、兰、虹、山、浩、中、牧	14.10%

表 3.5　高频双名首字用字及所占比例度

Tab. 3.5　High frequency first name in double name and proportion

双名首字序号	双名首字用字	所占比例
1-10	泽、镕、建、小、永、家、国、志、文、晓	22.75%
11-50	玉、明、德、大、洪、光、锦、长、克、正、岚、振、庆、瑞、秀、学、其、春、新、邦、海、世、金、健、铁、俊、伟、登、浩、立、福、万、广、忠、丽、宝、云、兴、兆、维	29.73%
51-100	启、全、天、红、东、荣、嘉、恩、华、成、思、淑、继、亚、昌、中、雪、怀、凤、培、树、宗、少、卫、元、宏、爱、一、敬、英、智、景、伯、桂、清、剑、子、仁、占、鲁、炳、杼、秋、纪、鸿、关、道、有、可、厚	15.33%

表 3.6　高频双名尾字用字及所占比例度

Tab. 3.6　High frequency last name in double name and proportion

双名尾字序号	双名尾字用字	所占比例
1-10	民、华、平、基、清、国、东、生、明、林	24.13%
11-50	辉、涛、云、杰、英、红、文、军、志、光、田、伟、春、新、山、琛、行、敏、宝、璇、中、庆、仁、祥、刚、龙、忠、莲、环、成、强、荣、才、江、良、峰、海、昌、丽、波	26.08%
51-100	安、梅、兴、顺、珍、元、亮、兰、萍、武、年、正、宁、芳、宇、德、根、来、贵、义、川、福、立、斌、松、群、康、功、玉、胜、映、权、坤、诚、远、滨、勇、彬、鹏、霖、章、禹、泉、玲、飞、森、农、喜、日、有	15.76%

3.2.3 上下文特征

在文本中，出现在人名前后的词称为上下文特征。人名在文本中表示一个具体的实体，其上下文信息和实体有关，或者说上下文在不同程度上表示人名的身份或者动作等特征。通过上下文特征可以很好的确定人名的边界，因此上下文特征对人名识别的定位和边界的确定有着至关重要的作用。

我们对文本中经常出现的上下文特征进行了总结，主要归纳为以下几类：

（1） 句首与句尾特征

人物通常作为某个事件的发动者、执行者、被讨论的对象，因此人名经常出现在句首充当主语或者出现在句尾充当宾语，例如“恩佐在讲话中对两国经贸合作业已取得的成就给予了高度评价”以及“叶利钦是在接受莫斯科新闻媒体的联合采访时发表上述看法的”。

统计数据显示人名前词与后词都出现 102731 次，去重后的人名前词共 4042 个，人名出现在句首时，前词使用 NULL 表示，NULL 出现次数为 30011 次，占所有前词出现次数的 29.21%，这是一个非常大的比重。去重后的人名后词为 5853 个，人名出现在句尾时，人名后词使用 NULL 表示，NULL 出现 10969 次，占所有后词出现次数的 10.68%。

（2） 表示称谓、职称、头衔的词

“记者”、“部长”、“主席”、“书记”、“总统”、“主任”、“老师”等都属于这类词。统计数据中部分词作为前词或者后词的出现次数如表 3.7 所示。

由表 3.7 可以看出，“记者”、“领导人”通常在人名前出现，“老师”、“同志”通常在人名后出现，而“主席”、“总理”在人名前后出现的次数相当。因此，有的称谓词适合作人名前词，有的称谓词适合作人名后词，有的称谓词既适合作人名前词又适合作人名后词。

表 3.7 称谓词作为前词或者后词出现次数

Tab. 3.7 The number of appellation occurrences before and after word

称谓词	作为前词出现次数	作为后词出现次数
记者	9292	1
领导人	178	1
老师	4	104
同志	19	2406
主席	2063	1818
总理	1226	799

（3）表示连接的词，例如“、”、“和”、“与”

多个人名经常连续出现，使用“、”、“和”、“与”进行分隔，表示人名的并列关系。统计数据显示“、”在人名后词中出现的次数最多，高达 10805 次。例如“吸纳了各方面学者专家谷源洋、谈世中、肖炼、李长久、陈漓高、甄炳禧等对世界经济形势的一些观点和分析。”。

（4）表示动作的词常作为人名后词

人名作为事件的主导者和执行者，其后经常跟一些动作表明人物的具体行为。例如“说”、“报道”、“对”、“指出”等词。例如“本报北京 1 月 5 日讯记者费伟伟报道”。

（5）常与人名一同出现的介词

这类词包括“除了”、“在”、“由”、“于”等。例如“贝尔曼此次来华演出的曲目除了穆索尔斯基的《图画展览会》”。

（6）“的”作为前后词

“的”作为人名前词，用来连接修饰词与人名。例如：“已是花甲之年的小戈麦尔斯基重出江湖，意欲重振当年的雄风”。“的”还可以作为人名后词，表示人物的某一物件、思想等。例如：“张某的营业房也未建在 310 国道 602 公里 1000 米处”。

上下文特征不仅对人名识别具有重要的指示作用，同时还给人名边界界定提供了强有力的依据。

3.3 本章小结

本章对中文人名识别任务进行了全面的分析，详细介绍了中文人名识别的意义，对中文人名识别的难点进行了总结，通过对 2000 年《人民日报》语料中的人名、前后词等信息的统计分析，对人名的姓氏特征、名用字特征、上下文特征进行了详细的总结。从人名的姓氏以及名用字特征来看，中国人名和日本人名中的姓氏以及名用字虽然比较分散，但大多还是集中在少数姓氏或者名用字上。而外国音译和少数民族音译人名用字比较分散，没有规律，需要依靠句法以及上下文信息进行识别，此外上下文特征对人名的识别和边界界定具有重要的指示作用。

4 基于循环神经网络的中文人名识别

中文人名主要包括中国人名、日本人名、外国音译、中国少数民族音译人名，目前大多数研究主要集中在中国人名的研究上，且取得了不错的成果，然而对日本人名、外国音译人名、中国少数名族音译人名研究力度不足。本文借鉴中文人名识别的研究经验，将中文人名识别问题转化为序列标注问题。而当今深度学习在自然语言处理领域取得了很好的效果，其中循环神经网络在序列标注问题上表现较好。因此，本文提出了一种基于循环神经网络的中文人名识别方法，循环神经网络无需花费大量的时间去挑选复杂的特征，通过对训练语料简单的训练即可得到人名识别模型，除此之外，针对训练语料不足的情况，还可以通过大量的未标注语料训练的词向量替代循环神经网络模型中的随机初始词向量，通过使用富含丰富语法语义的词向量作为模型训练的初始条件，可以很大程度的提高模型训练的质量。

该方法主要分为模型构建阶段和后处理阶段两个部分。在模型构建阶段的主要操作包括对初始语料进行分词、序号化、词转换为标签、数字泛化以及词向量的训练等操作，并整理为循环神经网络模型的输入进行中文人名识别模型的构建，并通过随机初始词向量的替换以及词向量的改进对模型进行改造；后处理阶段主要将测试语料处理后输入模型，得到识别结果，并将识别结果转化为人名候选，使用规则对候选人名进行筛选，通过正向逆向匹配算法提高人名识别的准确率，利用基于篇章的扩散操作对未识别的人名进行召回。

4.1 模型构建阶段

模型构建阶段主要包括五部分：语料预处理、词向量的训练、数词泛化、改进的 word2vec 以及模型训练等操作。其中语料的预处理主要处理语料，将语料处理为循环神经网络模型的输入形式；词向量的训练主要通过 word2vec 工具对词向量进行训练以及调整；数词泛化主要对数词进行统一化处理；改进的 word2vec 将特征信息融入到词向量中；模型训练阶段将处理好的语料输入到循环神经网络模型进行训练，经过多次迭代之后，得到收敛的模型。模型构建阶段流程如图 4.1 所示。

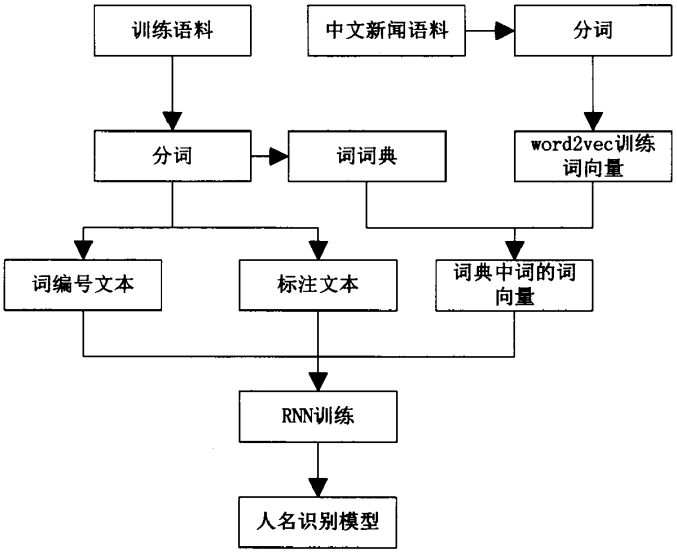

图 4.1　人名识别模型结构框图

Fig. 4.1　The structure diagram of　names recognition

4.1.1　语料预处理

首先对语料进行分词处理，本文选取的分词工具为本实验室罗彦彦[41]开发的基于 CRFs 边缘概率的分词系统。

由于在循环神经网络模型中，模型并不直接对词进行处理，而是将词转化为序号，通过序号与词向量矩阵 $V \in R^{m \times n}$ (其中 m 表示语料中词的总个数加 1，n 表示向量的维数)中的词向量联系到一块，其中 k 号即对应词向量矩阵的第 k 行，k 取值范围为[0,m]。因此分词之后，要对词进行序号化。首先对语料进行分词，建立词词典，使每一个词对应一个数字序号，由于训练语料的词是有限的，为了保证词典可以表示训练集、开发集、测试集中所有的词，词典从 1 号对出现的词进行编号，0 号保留用来表示在不存在与词典中的词，通过词词典即可将语料序号化。

在模型的构建过程中为了区分人名和非人名信息，本文采用 IOB2 的标注集合来标记人名，其中 B 表示人名的开始词，I 表示人名的非开始词，O 表示非人名的词。例如在中国人名中，B 表示姓，I 表示名，O 表示非中国人名的词。

本文将分类分为 6 类，其分类和标注如表 4.1 所示：

表 4.1 分类及标签信息

Tab. 4.1 The information of the classes and the labels

分类	标签
中国姓氏	B-C
中国名字	I-C
日本姓氏	B-J
日本名字	I-J
音译人名	B-F
非人名	O

采用表 4.1 中的分类即可根据标注文本将分词文本转换为标签文件，以“清朝著名学者郭嵩焘曾说”为例，首先对其进行分词，得到“清朝/著名/学者/郭/嵩/焘/曾/说”，然后其进行序号化以及转换为标签，如表 4.2 所示，其中词语列是经过分词工具分词得到的词，序号列是通过词词典将词转换为序号，IOB2 列则是根据分类信息为每一个词语进行人名标注。

表 4.2 人名标注举例

Tab. 4.2 The example of the names tagging

词语	序号	IOB2 标注
清朝	7	O
著名	9	O
学者	32	O
郭	17	B-C
嵩	65	I-C
焘	4	I-C
曾	22	O
说	11	O

从序号列可知，词“清朝”在词词典中的下标为 7，“学者”在词词典中的下标为 32，依次类推。从 IOB2 标注列可知“郭嵩焘”为中国人名。

4.1.2 词向量的训练

在神经网络中，词一般使用词向量进行表示，而神经网络模型在训练初始化参数过程中，只会对词向量进行随机初始化，由于词向量是随机初始化，初始状态下不带有任

何信息，此外如果训练语料过小，模型学习到的知识和信息就会过少，导致模型的效果不佳。

由于现今公开的标注语料集比较少，个人手工标注又不具有权威性，这就导致我们使用的标注语料非常有限，因此单纯的通过添加训练语料来提升系统的性能是非常困难的。如何使用有限的标注语料集来提升系统的性能就成为了重要的研究课题之一。

在神经网络模型中，使用词向量表示词，而词向量是可以事先通过大规模的无标注语料训练得到，同时词向量中还会包含大规模语料集中的句法、语义等丰富的信息。因此本文使用大规模无标注语料训练得到的词向量去替换神经网络模型中的初始词向量，通过此操作，神经网络模型在初始阶段，词向量就已经包含了丰富的信息，模型在已知丰富信息的前提下，接收训练语料进行模型的训练可以大大的提高系统的性能。Ronan Collobert 和 Jason Weston[42]将词向量用于自然语言处理领域的各项任务中，例如词性标注、命名实体识别、短语识别、语义角色标注等任务，实验结果表明使用词向量作为语言模型的初始值代替语言模型的随机初始值，其效果会有显著的提升。

本文采用 Google 开源词向量工具 word2vec 进行词向量的训练，该工具可以通过 https://code.google.com/p/word2vec/下载使用，以 2000 年《人民日报》新闻语料作为词向量的训练语料，将训练语料进行分词并送入 word2vec 中进行训练，其中，词向量的维度定为 100 维，训练窗口大小为 5，采样阈值为 12，模型采用 Hierarchical Softmax 实现的 Skip-gram 模型。

在向量空间中， 两个词的相似度可以通过空间的距离来判断，距离越近说明相似度越高，反之，相似度越低。距离指的是两个向量的余弦值，通过余弦值的大小表示两个词的相似度。使用距离信息检验词向量训练的效果。以姓氏“刘”为例，挑选出与词“刘”距离最近的词，并展示其相似度，如表 4.3 所示。

由表 4.3 可以看出在与词“刘”相近的词中，大部分是姓氏，例如：“王”、“陈”、“杨”、“张”、“郑”、“徐”等等，但是其中词“斌”、“树义”、“永年”、“卫东”、“淑”、“勇”等并非姓氏用字，我们可以从原语料中观察出，这些词都是人名的名字部分。因此使用 word2vec 训练获得的词向量主要受到其上下文信息的影响。

表 4.3 与“刘”最近的词及其相似度

Fig. 4.3 The words of the most similar to “刘” and its similarity

词	相似度	词	相似度	词	相似度
王	0.8352523	孙	0.73808515	树义	0.71722794
陈	0.8170227	薛	0.737594	永年	0.7170327
杨	0.80878574	沈	0.73658943	俞	0.71491694
张	0.7927441	崔	0.7344986	卫东	0.7096374
郑	0.782873	谭	0.7341375	淑	0.7083592
徐	0.7686511	范	0.73140025	蔡	0.70793563
赵	0.7634958	冯	0.7210508	袁	0.7042831
斌	0.74465346	翟	0.7178231	勇	0.7038869

在循环神经网络模型中，由于词向量矩阵 $V \in R^{m \times n}$ 随机初始化，矩阵的每一行都是随机的，因此语料中的词编号可以在 0 至 m 的范围内随意编号即可。而在使用训练好的词向量替换随机初始词向量时，每一个词向量都与每一个词相对应，这也就决定了词向量矩阵中第 k 行的词向量对应词词典中的 k 号词。只有词向量矩阵与词词典编号相对应了，才可以使系统正确执行。词向量矩阵与词编号以及词词典的关系如图 4.1 所示：

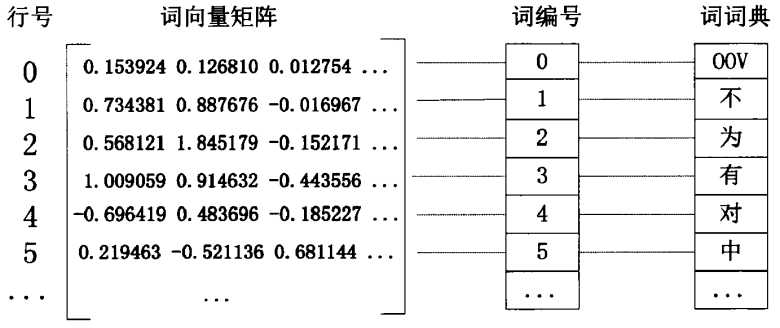

图 4.1 词向量矩阵与词编号以及词词典关系图

Fig. 4.1 The relationship between word embedding matrix, word number and word dictionary

由图 4.1 可知词向量矩阵的第 0 行向量对应词编号为 0 号，表示“OOV”这个词，“OOV”代表未登录词用来表示词词典中不存在的词。词向量矩阵的第 1 行向量对应词编号为 1，表示“不”这个词，依此类推。

4.1.3 数词泛化

由于中文词的个数不固定，字与字成词比较随意，加上词向量的训练可以通过设定阈值过滤掉出现频率不高的词，因此不管训练词向量的语料多大，都不可能将所有的词都包含在词向量词典中。然而如果语料中存在太多词向量词典中没有的词，其词向量还是要随机初始化，那么替换为已经训练好的词向量的作用就会相对减小。为了降低语料中不存在于词向量词典中词的个数，除了通过扩大词向量的训练语料外，还可以通过对数词进行泛化操作来缓解此类问题。

在语料中存在很多的数词，比如“11”、“34”、“56”、“一九九八”，而这些词在语料中比较多，且没有实际的意义，在人名识别领域，数词是“11”还是“12”对人名识别毫无影响。本文针对数词的影响做了如下统计，如表 4.4 所示：

表 4.4 数词在各语料中所占的比重

Fig. 4.4 The weight of the numerals in the corpus

词集合	总词数	数词出现次数	数词所占比例
训练语料	958249	21221	2.21%
测试语料	232310	4810	2.07%
存在于训练语料而不存在于词向量文件的词集合	3907	1806	46.22%

由表 4.4 可知，数词在训练语料和测试语料中均占到 2%的比重，但是在没有词向量的词集合中数词所占的比重却高达 46%，因此，数词出现的情况比较多，对数词进行统一为同一个数字即可减少数词的影响。

本文对数词的处理的主要思想为：对分词语料的每一个词进行遍历，如果词仅由阿拉伯数字、百分号（%）、点（.）、连接符（-）、斜杠（/）构成，则将此词替换为数字“1”，如果词仅由“零”、“一”、“二”、“三”、“四”、“五”、“六”“七”、“八”、“九”、“十”、“百”、“千”、“万”、“亿”组成，且词长度大于 2，并且词不能以“万”开头，则将词替换为数字“1”。对于中文的数字表示，此处考虑到人名中存在数词的情况，且“万”可以作为姓氏出现，因此在数词泛化时，通过词长度信息和首字不能为“万”字作为限制条件。经过这样的操作，像人名“万三千”、“张三”等不会因为数词的泛化对人名识别产生影响。

4.1.4 改进的 word2vec

Google 开源的 word2vec 工具在对分词语料进行词向量训练过程中，仅仅利用词的上下文信息，使上下文信息一致的词在向量空间中距离相近，从而通过向量空间的距离来反应词与词之间的相似程度。然而使用 word2vec 进行词向量的训练时，存在如下问题：

（1）使用高质量、大规模的分词语料训练词向量，可以提高词向量的质量。但是由于中文的随意性，语料不可能涵盖所有词的用法，同时网络中爬取的中文文本形式多样、标准不一，高质量的中文文本不容易获取。

（2）词向量对上下文信息依赖较大，通过相似度计算得到的相似词大多为其上下文信息。从表 4.3 中既可以看出，与姓氏“刘”相似的词中包含了“斌”、“树义”、“永年”、“卫东”、“淑”、“勇”等词，这些词均为与词“刘”共现的词。

为了缓解上述问题，训练更高质量的词向量，本文对 word2vec 进行改进，其改进思想为：首先对中文语料进行分词，提取词的特征，并将特征存入到特征文件中，然后使用 word2vec 对分词语料进行训练，得到词向量 $\ell 1$ ，然后将一个特征文件作为改进后的 word2vec 的输入进行 $\ell 1$ 的更新得到融入特征的词向量 $\ell 2$ ，使用改进后的 word2vec 依次对其他特征进行训练，最终得到融入所有特征的词向量。

与 word2vec 不同之处在于，在对特征进行训练时，不再是对特征的词向量进行更新，而是对原文件中的词进行词向量的更新，通过借助每个特征构建的霍夫曼树结构，将特征信息融入到原文本中词的词向量中。

以“迈向 充满 希望 的 新 世纪”为例进行说明，特征仅选择词性特征。

原文件	：迈向	充满	希望	的	新	世纪
词性特征	：COM-VERB	NVERB	NVERB	DE-1	ADJ-PRE	COM-NOUN

首先，使用 word2vec 对原文件进行词向量的训练，以训练“希望”的词向量为例进行说明，示意图如图 4.2 所示。

图 4.2 展示了 word2vec 训练词向量的流程，在霍夫曼树的根节点输入的是词“希望”的词向量 v(希望)，窗口大小定位 3，则需要预测 P(充满|希望)和 P(的|希望)这两个概率。此处以 P(充满|希望)为例进行说明，即已知“希望”的前提下预测其前后词为“充满”的概率。因此需要从根节点经过三个分类器到达词为“充满”的叶子节点，路径编码为 110，在每次二分类问题中，都会产生误差（被正确分类的概率*v(希望)），并将误差进行累计更新到 v(希望)的词向量中。当原文件中所有词都被训练之后，即得到了词向量 $\ell 1$ 。

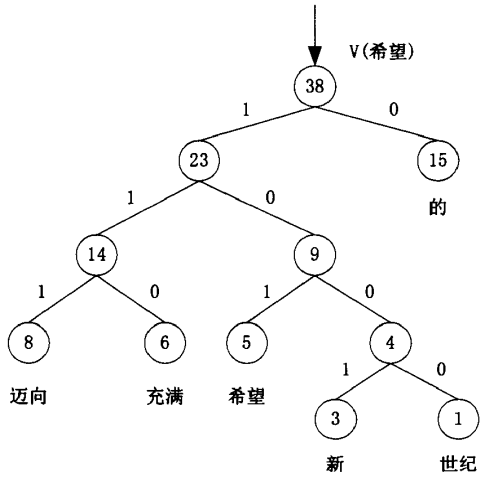

图 4.2　word2vec 模型训练词向量示例

Fig. 4.2　An example of training word embedding with the word2vec

然后使用改进的 wrod2vec 对特征文件进行训练，其训练流程如图 4.3 所示。

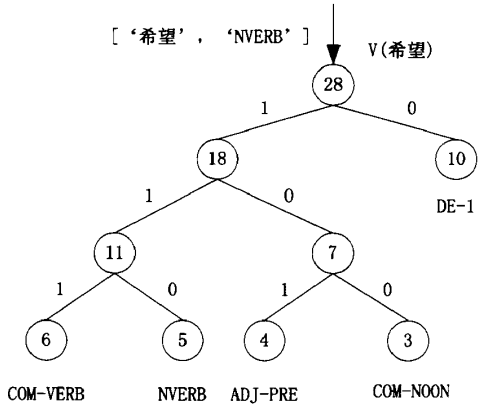

图 4.3　改进后 word2vec 模型训练词向量示例

Fig. 4.3　An example of training word embedding with the improved word2vec

由图 4.3 可知，霍夫曼树为词性特征的霍夫曼树，word2vec 无法通过词性特征去更新原文本的词向量。比如词“希望”所对应的词性为“NVERB”，应该预测 P(NVERB|NVERB)和 P(DE-1|NVERB)的概率，为了将词性特征的词向量更新到对应的

词中，本文将预测概率修改为 P(NVERB|希望)和 P(DE-1|希望)，以 P(NVERB|希望)为例进行说明，其中，根节点的输入不再是 v(NVERB)而是从 $\ell 1$ 中获取的词“希望”的词向量，从根节点经过三个分类器到达“NVERB”的叶子节点，将误差进行累计更新到 v(希望)的词向量中，这样即通过词性的霍夫曼树结构更新了原文件中词的词向量信息。当词性特征全部训练完成，即得到了融入词性特征的词向量 $\ell 2$。

词性是词最基本也是最重要的特征，因此本文仅引入词性特征，探究词性特征对词向量的影响。本文对 2000 年《人民日报》新闻语料进行分词和词性标注，并将分词与词性信息分离，并使用改进后的 word2vec 模型进行词向量的训练，词向量训练参数与 4.1.2 节中参数保持一致。仍然以“刘”为例，比较 word2vec 与改进后的 word2vec 的词相似性，如表 4.5 所示：

表 4.5　wor2vec 与改进后 word2vec 词相似性对比

Tab. 4.5　Comparison word similarity between the word2vec and the improved word2vec

word2vec				改进后的 word2vec			
词	相似度	词	相似度	词	相似度	词	相似度
王	0.8352523	谭	0.7341375	王	0.8811034	朱	0.7759748
陈	0.8170227	范	0.73140025	陈	0.8721117	谭	0.76571566
杨	0.80878574	冯	0.7210508	李	0.8705164	崔	0.75605255
张	0.7927441	翟	0.7178231	赵	0.8506992	沈	0.7463815
郑	0.782873	树义	0.71722794	杨	0.8418417	冯	0.74575084
徐	0.7686511	永年	0.7170327	张	0.825872	汪	0.7445462
赵	0.7634958	俞	0.71491694	郑	0.8158898	曹	0.74289554
斌	0.74465346	卫东	0.7096374	吴	0.8121026	潘	0.7282312
郭	0.74031055	淑	0.7083592	孙	0.80823517	宋	0.71014047
孙	0.73808515	蔡	0.70793563	胡	0.7979869	范	0.7100426
薛	0.737594	袁	0.7042831	徐	0.79057527	蒋	0.70557225
沈	0.73658943	勇	0.7038869	郭	0.77996784	黄	0.7046592
崔	0.7344986	尹	0.7035962	袁	0.7773723	俞	0.70280606

由表 4.5 可以看出，两种模型中，改进的 word2vec 训练获得的词向量具有更好的效果，不仅提高了姓氏之间的相似度，而且在相似度较高的词中，类别也较为统一，在挑选出的 26 个词中，word2vec 模型中存在“斌”、“树义”、“永年”、“卫东”、“淑”、“勇”6 个非姓氏词，在改进后的 word2vec 则全为姓氏词。

4.1.5 模型训练

使用训练好的词向量、标注文本、序号化文本作为循环神经网络模型的输入，词向量主要用于存储词的相关信息，其包含的信息可以直接影响模型的性能；标注文件主要用于提供真实值，使用真实值与预测值的误差评定参数的质量，如果误差过大，则需要采用 BPTT 算法进行逆向参数修正；序号化文本用于模型的训练，为模型训练提供数据，并通过序号与词向量矩阵建立联系。

在使用循环神经网络进行训练前，需要将训练数据的顺序打乱，打乱顺序进行训练的好处在于使各种标签类型的案例分布均匀，从而不会出现一直训练一种标签类型案例的情况。通过将训练数据均匀分布可以使训练的模型更快的达到收敛状态。Mikolov[43] 指出通过对训练语料使用随机次序进行训练，可以有效的减少训练所需的迭代次数。

4.2 后处理阶段

将测试语料进行分词，使用模型构建阶段的词词典对测试语料的分词语料进行序号化操作，然后将序号化之后的分词语料输入到已经训练好的中文人名识别模型中，经过模型的训练，输出识别结果，最后对识别结果进行后处理操作。

人名识别阶段的具体流程如下所示：

步骤 1：人名识别模型对测试数据进行识别，提取识别后的分类标签并将相应的标签转化为人名字符串，作为候选人名。

步骤 2：使用上下文规则筛选候选人名，过滤不符合规则的人名。

步骤 3：去除长度为 1 的候选人名，将筛选后的候选人名定为最终人名，并使用基于篇章的全局扩散算法召回已经识别出而在上下文信息不足或者上下文信息过拟合的位置中未被识别的人名。

步骤 4：使用基于篇章的局部扩散算法召回有名无姓、有姓无名的人名。

具体识别流程如图 4.4 所示：

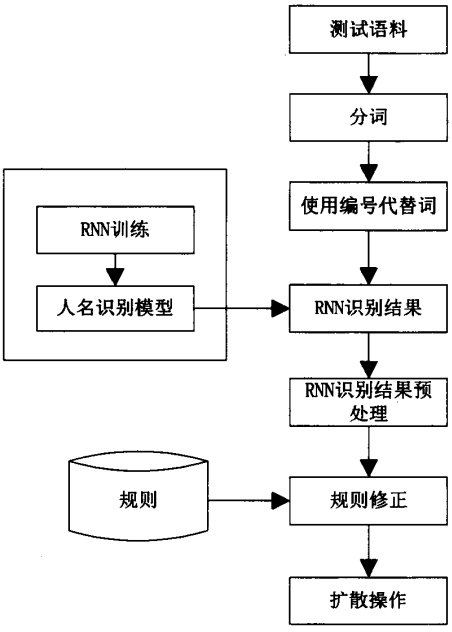

图 4.4　人名识别及其后处理流程图

Fig. 4.4　The flow chart of names recognition and post-processing

4.2.1　上下文规则

通过规则可以有效的过滤识别不正确的人名，部分使用规则实例如下：

（1）过滤修饰性的词。比如：老、小等；

（2）过滤称谓词。比如嫂、祖父、父；

（3）建立人名停用词词表，存在则过滤。

通过规则筛选，可以有效的去除经常与人名搭配，而又不能作为人名一部分的词，比如“小”字不能出现在人名的第一个字，否则“小”就不是名字的一部分，应过滤。这样可以将错误人名过滤掉，避免产生级联错误。通过上下文规则的过滤操作，可以为后续的工作创造好的识别环境，同时此项工作也是后续处理的基础，具有重要的意义。

4.2.2　全局扩散和局部扩散

在同一篇章中，同一人名经常在不同上下文语境中多次出现，在上下文特征明显的地方模型比较容易识别，而在上下文特征不明显或者特征过拟合的地方会出现模型识别

不出的情况。例如：在句子“马克思/的/历史唯物主义/认为”中“马克思”可以识别出来，而在句子“试图/发掘/马克思/理论/的/意义”中“马克思”的含义更偏向于一种理论而非人名，没有被识别出。基于人名的这种特征，采用全局扩散的思想解决此类问题。

全局扩散主要思想为：使用识别出来的人名遍历整个篇章，如果存在完全一致的字符串，则将匹配到的字符串当作人名。

同时，人名也经常以只有姓氏或者只有名字的形式出现，比如在句子“罗/建军/曹/均/宏”和“建军/，/建军/，/你/可是/稀客/ 呀/！”中，人名“建军”就是“罗建军”的名字部分。对于此类问题，采用局部扩散的思想解决。

局部扩散主要思想为：如果存在只有姓氏或者只有名字的人名 name，那么 name 附近一般存在 name 的全名。提取出整个篇章的名字信息，一次遍历篇章中的人名，检查是否存在人名包含 name，如果包含则将 name 确定为人名，否则从人名候选中剔除。

通过基于篇章的全局扩散和基于篇章的局部扩散操作，可以有效的提高人名识别的召回率。

4.3 本章小结

本章主要对系统实现进行了详细的介绍，将系统的实现划分为模型构建阶段和人名识别阶段两部分，在模型构建阶段主要介绍了对语料的前期处理，以及对词向量的操作，然后对模型的训练细节进行了说明。在人名识别阶段主要介绍了后处理的整个流程，首先通过规则过滤提高人名识别的准确率，然后使用基于篇章的全局扩散与局部扩散操作提高人名识别的召回率。

5 实验

5.1 数据集说明

本文使用1998年《人民日报》新闻语料作为数据集，并将数据集分为15份，按照11:1:3的比例切分为训练集、开发集、测试集。训练集用于训练人名识别模型，开发集辅助模型进行参数调优，测试集用于测试模型性能。

数据集中包含中国人名，日本人名，音译人名，此外还包括有名无姓、有姓无名、笔名等特殊中文人名的情况。其中，训练集中包含人名11927个，开发集中包含人名3528个人名，测试集中包含人名3512个。

5.2 实验设计

为了对系统进行全面的测试和分析，本文设计三个实验分别从三个方面对系统进行测试，首先我们将循环神经网络模型中的词向量矩阵作为对象，研究词向量矩阵中词向量的质量对模型训练的影响，其次对系统的后处理操作进行有效性测试，最后与其他研究方法进行对比。

5.2.1 循环神经网络模型性能实验

该实验主要以循环神经网络模型中的词向量矩阵作为研究对象，通过对其进行优化，来分析其对循环神经网络模型性能的影响。为了验证此实验，我们进行了如下4组实验，详细实验设计如下：

（1）使用循环神经网络模型中的随机初始词向量进行中文人名识别模型的训练。将此实验记为RNN。

（2）使用word2vec对2000年《人民日报》新闻语料进行训练得到词向量，并使用富含丰富语义信息的词向量替换神经网络模型中随机初始词向量，然后进行中文人名识别模型的训练。将此实验记为Emb。

（3）为了消除中文文本中数词的影响，对本文所使用到的数据进行数词泛化。首先对2000年《人民日报》新闻语料进行泛化，然后使用word2vec进行训练得到词向量，使用该词向量替换循环神经网络中的随机初始词向量，训练中文人名识别模型。将该实验记为Digit。

（4）使用改进的 word2vec 模型对泛化后的 2000 年《人民日报》新闻语料进行训练，得到融入词性特征的词向量，并替换循环神经网络模型中的随机初始词向量，进行中文人名识别模型的训练。将该实验记为 ReEmb。

对以上 4 组实验进行 20 次迭代，使用开发集对每次迭代的模型进行进行评估，并根据评估结果自调节学习率，以期在开发集上获取更好的效果。4 组实验结果如表 5.1 所示。

表 5.1 实验 F 值结果

Tab. 5.1 The F values of experiments

迭代次数	RNN-T	RNN-V	Emb-T	Emb-V	Digit-T	Digit-V	ReEmb-T	ReEmb-V
0	0.7361	0.7850	0.7089	0.7524	0.7276	0.7648	0.7746	0.8348
1	0.7802	0.8629	0.8019	0.8764	0.8101	0.8802	0.8029	0.8454
2	0.7916	0.8276	0.8299	0.8748	0.8314	0.8864	0.8311	0.9064
3	0.8021	0.8645	0.8319	0.8975	0.8321	0.9035	0.8149	0.9104
4	0.8047	0.8629	0.8213	0.8435	0.8368	0.9044	0.8469	0.9047
5	0.7965	0.8997	0.8313	0.9026	0.8310	0.8945	0.8542	0.8935
6	0.8121	0.8400	0.8203	0.9045	0.8297	0.8891	0.8395	0.8943
7	0.8155	0.8952	0.8366	0.8893	0.8358	0.9012	0.8428	0.9122
8	0.8317	0.8894	0.8493	0.8825	0.8507	0.9045	0.8634	0.9135
9	0.8329	0.8830	0.8343	0.8844	0.8487	0.9071	0.8735	0.9293
10	0.8415	0.8983	0.8408	0.8876	0.8496	0.8991	0.8728	0.9294
11	0.8428	0.8864	0.8524	0.9053	0.8554	0.9120	0.8738	0.9305
12	0.8490	0.8996	0.8579	0.914	0.8581	0.9167	0.8736	0.9311
13	0.8545	0.9044	0.8585	0.9145	0.8601	0.9193	0.8727	0.9303
14	0.8542	0.9052	0.8573	0.9144	0.8604	0.9187	**0.8740**	0.9311
15	0.8536	0.9069	0.8591	0.9163	0.8617	0.9207	0.8726	0.9294
16	0.8506	0.9077	0.8553	0.9177	0.8603	0.9209	0.8744	0.9306
17	0.8512	0.9082	**0.8566**	0.9190	0.8598	0.9199	0.8751	0.9301
18	**0.8517**	0.9084	0.8568	0.9176	**0.8608**	0.9210	0.8755	0.9310
19	0.8511	0.9058	0.8550	0.9186	0.8605	0.9214	0.8749	0.9307

其中，T 代表测试集，V 代表开发集。从结果可以看出使用 word2vec 训练得到的词向量替换循环神经网络随机的词向量，其结果的 F 值要比单纯使用循环神经网络的 F 值高 0.5%左右，进行数词泛化后 F 值又提高了 0.4%左右，使用改进的 word2vec 获得的词向量，其 F 值提高了 1.1%左右。取得了较好的成果。

为了能够展现实验迭代次数对实验结果的影响，将表 5.1 的数据绘制成折线图如图 5.1 所示。

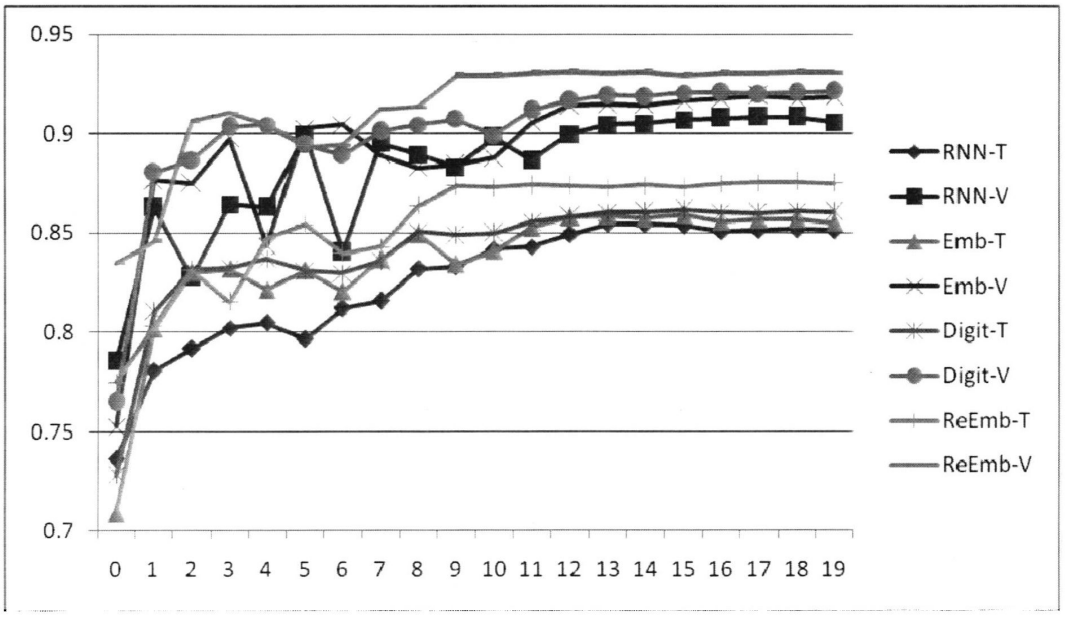

图 5.1　基于循环神经网络模型的中文人名识别方法实验折线图

Fig. 5.1　Experimental line chart of Chinese name recognition method based on RNN

由图 5.1 可以看出，在前 10 次迭代过程中，F 值波动较大，在第 15 次迭代之后 F 值趋于稳定，模型收敛。此外，还可以发现，词向量中包含的信息越丰富，其 F 值波动范围就会波动越小，越早的趋于收敛。

我们选取收敛之后的模型作为我们训练得到的最终模型，由于迭代 15 次之后的模型趋于收敛，这里将第 15 次迭代之后在开发集上表现最好的一次迭代得到的中文人名识别模型作为最终的模型。由此四组实验的准确率、召回率、F 值如表 5.2 所示。

由表 5.2 可以看出，通过不断的丰富循环神经网络中的词向量信息，中文人名识别模型在准确率、召回率以及 F 值上都有所提升，分别提高了 2.82%、1.68%、2.23%。由此可见，通过优化词向量的质量可以使循环神经网络模型学习到更多有益的信息，提高系统的性能。

表 5.2 模型构建阶段实验结果

Tab. 5.2 The experimental results of model construction

方法	P	R	F
RNN	86.29%	84.08%	85.17%
Emb	86.88%	84.48%	85.66%
Digit	87.57%	84.65%	86.08%
ReEmb	89.11%	85.76%	87.40%

5.2.2 验证后处理方法的有效性

本实验首先处理人名识别模型的识别结果，主要是将人名识别模型标记的 B、I、O 标签转换为对应的字符串，并将连续的 B、I 标签组成人名字符串。然后使用规则筛选掉不能成为人名的字符串。经过规则处理可以保证人名字符串的准确性。最后使用基于篇章的全局扩散算法在篇章中匹配，如果存在完全匹配的字符串则取出作为人名字符串。通过基于篇章的全局扩散可以召回模型已经识别出，但在某些上下文信息不足的位置未被识别的人名。在中文人名中对于使用名字的一部分作为人名的情况，通过基于篇章的局部扩散方法进行召回。通过基于篇章的全局和局部扩散操作可以提高人名识别的召回率。

依次对中文人名识别的结果加入规则过滤、全局扩散以及局部扩散等操作，通过实验数据观察人名识别的有效性，实验结果如表 5.3 所示：

表 5.3 后处理阶段实验结果

Tab. 5.3 The experimental results of post processing

方法	P（准确率）	R（召回率）	F 值
ReEmb	89.11%	85.76%	87.40%
+规则	92.02%	85.32%	88.54%
+全局扩散	92.83%	88.86%	90.80%
+局部扩散	93.06%	91.45%	92.24%

实验数据显示，在添加规则后人名识别的准确率由 89.11%上升到 92.02%。由此可以看出规则过滤操作主要是在人名的准确性上做出了大的贡献。其次通过全局扩散和局部扩散操作，人名的召回率从 85.32%升到了 91.45%，由此可以得出扩散操作有利于人名的召回。综上，基于循环神经网络的中文人名识别方法是有效的。

5.2.3 对比实验

为了验证系统的性能，使用同文献[44]和文献[45]同样的训练集与测试集进行对比实验。其中文献[44]利用多级阈值的方式识别中文人名，该方法使用统计的方法对大规模语料进行分析，并从中提取出中文人名的姓氏、人名用字等规律，提出了针对汉族人名、少数民族人名以及外国音译人名的多级阈值概念，通过确定阈值进行中文人名的识别。文献[45]利用机器学学习的方法对人名进行识别，提取多种特征，并编写特征模板进行 CRF 模型的训练，并定义多种人名可信度模型对人名进行识别。其对比结果如表 5.4 所示：

表 5.4 与其他方法进行对比

Tab. 5.4 The comparison with other experiments

方法	P（准确率）	R（召回率）	F 值
文献[44]	86.71%	92.49%	89.50%
文献[45]	92.80%	90.60%	91.70%
本文方法	93.06%	91.45%	92.24%

文献[44]在处理人名时没有考虑有姓无名、有名无姓以及笔名等特殊情况，且对于较长的人名识别效果欠佳，本文通过局部扩散操作对有姓无名、有名无姓的人名进行了召回，且在识别过程中对人名长度没有限制。文献[45]在构建模型过程中抽取特征操作比较复杂，本文仅使用词向量作为特征，且词向量可以随着训练语料的增加学习到更多更丰富的语义信息，因此取得了较好的效果。

5.3 本章小结

本章主要介绍了本文所使用的语料以及实验设计。为了证明系统的有效性，本文进行了 3 组实验，首先对循环神经网络模型的词向量矩阵进行了 4 个实验，验证了词向量质量与模型性能之间的关系，然后对系统的后处理的有效性进行了实验分析，最后与其他的中文人名识别系统进行了对比实验，由此证明我们的系统是有效的。

结　论

中文人名包括中国人名、日本人名、外国音译人名以及少数民族音译人名，涵盖范围较广。此外，其随意性使其在未登录词中占有较大的比重，解决未登录词识别问题首先要解决中文人名识别问题。因此，中文人名识别任务是中文信息处理任务中的基础任务，其性能的好坏将直接影响到其他任务的性能，故其在自然语言处理领域具有重要的地位。

本文对中文人名识别任务进行了详细的分析以及概括，提出了一种基于循环神经网络的中文人名识别方法。相对于统计机器学习的方法，该方法无需花费大量的时间挑选复杂的特征，仅仅使用词向量作为模型特征，同时，该方法不需人工干预，避免了人工操作对实验造成影响。此外，针对训练语料不足的情况，还可以通过大量未标注语料训练得到的词向量替代循环神经网络模型中的随机初始词向量，通过富含丰富语义信息的词向量作为模型训练的初始条件，可以有效的提高模型性能。

本文主要针对循环神经网络模型中存储信息的词向量入手，通过不断的优化改进词向量，从而提高系统的性能。此外，还通过一系列的后处理操作提高系统的精确率与召回率，进一步提升了系统的性能。

尽管我们在中文人名识别任务中取得了较好的结果，但其识别的精度和广度还有待提高。因此我们下一步的工作应该考虑如何改进现有的系统，提高现有系统的性能。其次，中文语料中音译人名的相关语料较少，系统学习到的相关信息也就较差，从而导致音译人名识别不全等现象，我们应该加大对音译人名语料的收集。此外，我们还可以寻找更多的有益特征，将特征融入到词向量中供模型学习，期望能够取得更好的效果。

参 考 文 献

[1] Wang Z H H, Li S. The Task 2 of CIPS-SIGHAN 2012 Named Entity Recognition and Disambiguation in Chinese Bakeoff[J]. CLP 2012, 2012: 108.

[2] Carpineto C, Romano G. A survey of automatic query expansion in information retrieval[J]. ACM Computing Surveys (CSUR), 2012, 44(1): 1.

[3] Chiang D, Knight K, Wang W. 11, 001 new features for statistical machine translation [C]// Proceedings of Human Language Technologies: The 2009 Annual Conference of the North American Chapter of the Association for Computational Linguistics, 2009: 218-226.

[4] Prettenhofer P, Stein B. Cross-language text classification using structural correspondence learning [C]// Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics, 2010: 1118-1127.

[5] 黄德根，朱和合. 基于最长次长匹配的汉语自动分词[J]. 大连理工大学学报，1999, 39(6): 831-835.

[6] 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.

[7] 费洪晓，康松林，朱小娟，等. 基于词频统计的中文分词的研究[J]. 计算机工程与应用，2005, 41(7): 67-68.

[8] 张梅山，邓知龙，车万翔，等. 统计与词典相结合的领域自适应中文分词[J]. 中文信息学报，2012, 26(2): 8-13.

[9] 蒋建洪，赵嵩正，罗玫. 词典与统计方法结合的中文分词模型研究及应用[J]. 计算机工程与设计，2012, 33(1): 387-391.

[10] Chen X, Qiu X, Zhu C, et al. Gated recursive neural network for Chinese word segmentation[C]//Proceedings of Annual Meeting of the Association for Computational Linguistics. pendency parsing using two heterogeneous gated recursive neural networks. In Proceedings of the Conference on Empirical Methods in Natural Language Processing. 2015.

[11] 来斯惟，徐立恒，陈玉博，等. 基于表示学习的中文分词算法探索[J]. 中文信息学报，2013, 27(5): 8-15.

[12] Zheng X, Chen H, Xu T. Deep Learning for Chinese Word Segmentation and POS Tagging[C]//EMNLP. 2013: 647-657.

[13] Chen X, Qiu X, Zhu C, et al. Long short-term memory neural networks for chinese word segmentation[C]//Proceedings of the Conference on Empirical Methods in Natural Language Processing. 2015.

[14] 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.

[15] Zhao H, Liu Q. The CIPS-SIGHAN CLP 2010 Chinese word segmentation bakeoff[C]//Proceedings of the First CPS-SIGHAN Joint Conference on Chinese Language Processing. 2010: 199-209.

[16] Duan H, Sui Z, Tian Y, et al. The CIPS-SIGHAN CLP 2012 Chinese word segmentation on microblog corpora bakeoff[C]//Proceedings of the Second CIPS-SIGHAN Joint Conference on Chinese Language Processing, Tianjin, China. 2012: 35-40.

[17] 周昆，胡学钢. 一种基于本体论和规则匹配的中文人名识别方法[J]. 微计算机信息，2010 (31)：87-89.

[18] 李建华，王晓龙. 中文人名自动识别的一种有效方法[J]. 高技术通讯，2000，10(2)：46-49.

[19] Bei L I, Lei Z. 基于错误驱动学习和知网的中文人名识别[J]. 计算机工程，2012，38(12)：179-181.

[20] Gu C, Tian X, Yu J. Automatic Recognition of Chinese Personal Name Using Conditional Random Fields and Knowledge Base[J]. Mathematical Problems in Engineering, 2015.

[21] 曾辉，王俊，李艳. 基于 Naive Bayes 的中文人名识别研究[J]. 科学技术与工程，2015 (6)：83-86.

[22] 李丽双，黄德根，毛婷婷，等. 基于支持向量机的中国人名的自动识别[J]. 计算机工程，2006，32(19)：188-201.

[23] 黄德根，杨元生，王省，张艳丽，钟万勰. (2001). 基于统计方法的中文姓名识别. 中文信息学报，*15*(2)，32-38.

[24] 黄德根，马玉霞，杨元生. 基于互信息的中文姓名识别方法[J]. 大连理工大学学报，2004，44(5)：744-748.

[25] 王祖兴，吕 钊，顾君忠.基于混合方法的中文人名识别研究[J].计算机工程与应用，2015，51 (8):211-217

[26] 钱晶，张玥杰，张涛. 基于最大熵的汉语人名地名识别方法研究[J]. 小型微型计算机系统，2006，27(9)：1761-1765.

[27] 王玥，吕学强，李卓，等. 搜索日志中中文人名自动识别[J]. 中文信息学报，2015，29(3)：162-168.

[28] Williams D E R G E H R J, Hinton G E. Learning representations by back-propagating errors[J]. Nature, 1986, 323: 533-536.

[29] Tomas Mikolov, Quoc V. Le, Ilya Sutskever. Exploiting Similarities among Languages for Machine Translation. arXiv:1309.4168v1, 2013

[30] Mikolov T, Sutskever I, Chen K, et al. Distributed representations of words and phrases and their compositionality[C]//Advances in neural information processing systems. 2013: 3111-3119.

[31] Elman J L. Finding structure in time[J]. Cognitive science, 1990, 14(2): 179-211.

[32] Bengio Y, Schwenk H, Senécal J S, et al. Neural probabilistic language models[M]//Innovations in Machine Learning. Springer Berlin Heidelberg, 2006: 137-186.

[33] Williams D E R G E H R J, Hinton G E. Learning representations by back-propagating errors[J]. Nature, 1986, 323: 533-536.

[34] Mikolov T, Karafiát M, Burget L, et al. Recurrent neural network based language model[C]//INTERSPEECH. 2010, 2: 3.

[35] Mikolov T, Kombrink S, Burget L, et al. Extensions of recurrent neural network language model[C]//Acoustics, Speech and Signal Processing (ICASSP), 2011 IEEE International Conference on. IEEE, 2011: 5528-5531.

[36] Sutskever I, Martens J, Hinton G E. Generating text with recurrent neural networks[C]//Proceedings of the 28th International Conference on Machine Learning (ICML-11). 2011: 1017-1024.

[37] Liu S, Yang N, Li M, et al. A Recursive Recurrent Neural Network for Statistical Machine Translation[C]//ACL (1). 2014: 1491-1500.

[38] Sutskever I, Vinyals O, Le Q V. Sequence to sequence learning with neural networks[C]//Advances in neural information processing systems. 2014: 3104-3112.

[39] Auli M, Galley M, Quirk C, et al. Joint Language and Translation Modeling with Recurrent Neural Networks[C]//EMNLP. 2013, 3(8): 0.

[40] Graves A, Jaitly N. Towards end-to-end speech recognition with recurrent neural networks[C]//Proceedings of the 31st International Conference on Machine Learning (ICML-14). 2014: 1764-1772.

[41] 罗彦彦、黄德根. 基于 CRFs 边缘概率的中文分词[J]. 中文信息学报，2009，23(5)：3-8.

[42] Collobert R, Weston J. A unified architecture for natural language processing:Deep neural networks with multitask learning[C] //Proceedings of the 25th International conference on Machine learning. ACM, 2008: 160-167

[43] Mikolov T, Deoras A, Povey D, et al. Strategies for training large scale neural network language models[C]//Automatic Speech Recognition and Understanding (ASRU), 2011 IEEE Workshop on. IEEE, 2011: 196-201.

[44] 余祖波，高庆狮，马建军. 基于多级阈值的中文人名识别[J]. 计算机工程与应用,2007, 43 (33):1-3.

[45] 王祖兴，吕 钊，顾君忠.基于混合方法的中文人名识别研究[J].计算机工程与应用, 2015, 51 (8):211-217

攻读硕士学位期间发表学术论文情况

1. 一种基于循环神经网络的中文人名识别方法. 黄德根，**徐新峰**. 专利受理，申请号：201610308475.6 （本硕士论文第四章）
2. 基于广义 Jaccard 系数的汉语微博情感新词的判定. 桑乐园，**徐新峰**，张婧，黄德根. 山东大学学报（理学版），2015 年，50（7）：71-75. 主办单位：山东大学。

本文获得国家自然科学基金资助

1．国家自然科学基金（61173100）“跨语言信息检索中的机器翻译研究”。

致　谢

首先要感谢我的导师黄德根教授，从初出茅庐的懵懂与遇到问题的逃避，成长到现在对学术的热衷与遇到困难的面对，这都离不开黄老师的指导。黄老师在学术上的独特的见解与远见，指明了我学术的方向，并在生活中教会我自立自强的道理，使我在研究生生活中受益匪浅。

其次，感谢我的父母，尊重我的每次一决定，让我做自己喜欢做的事情。感谢父母无私的奉献与关怀，让我相隔千里仍时时挂念。感谢父母，为我永远保留着避风的港湾，让我免受伤害。

感谢我们实验室的全体成员，提供给我一个安静融洽的学习氛围。在老师的带领下，活跃的学习氛围使我对学术的研究乐此不疲。同时，正是因为有这样一个环境，才能让我顺利的完成科研工作。

感谢我的舍友，三年来的陪伴让我感受到家的温馨；感谢周广博、张建海、范蒙、万佳、田雪、桑乐园在生活上的谦让与工作上的支持，你们鼓励的话语是我奋进的动力，强有力的后盾支撑；感谢张建海在学术上的指导，孜孜不倦的教导总能起到画龙点睛的作用；感谢万佳在写作上的帮助以及生活中无微不至的照顾，让我能够将更多的精力放在科研工作上；感谢田雪、王冠群、叶子语，项目不单单激励着我们在技术上互相学习，还让我们更好的融为一个大家庭。

最后，再一次送上我由衷的祝福，感谢我的家人，朋友，同学，老师。

大连理工大学学位论文版权使用授权书

本人完全了解学校有关学位论文知识产权的规定，在校攻读学位期间论文工作的知识产权属于大连理工大学，允许论文被查阅和借阅。学校有权保留论文并向国家有关部门或机构送交论文的复印件和电子版，可以将本学位论文的全部或部分内容编入有关数据库进行检索，可以采用影印、缩印、或扫描等复制手段保存和汇编本学位论文。

学位论文题目：基于循环神经网络的中文人名识别的研究

作 者 签 名：徐新峰　　日期：2016 年 6 月 12 日

导 师 签 名：　　日期：2016 年 6 月 12 日