文章编号：1003-0077(2021)07-0041-06

基于大规模语料库的古文词典构建及分词技术研究

邢付贵[1,2]，朱廷劭[1,2]

(1. 中国科学院 心理研究所，北京 100101；
2. 中国科学院大学 心理学系，北京 100049)

摘 要：古文献的研究有助于传统文化的继承与发扬，而古文分词则是利用自然语言处理技术对古文献进行分析的重要环节。当前互联网拥有大量古汉语文本和词典方面的数据资料，该文提出利用互联网大规模古文语料构建古文基础词典；进而通过互信息、信息熵、位置成词概率多特征融合的新词发现方法从大规模古籍文本中建立候补词典；最终将基础词典与候补词典融合，形成含有 349 740 个字词的集成古文词典 CCIDict。在 CCIDict 基础上，利用多种分词算法实现古文的分词。基于 CCIDict 的正向最大匹配算法与开源的分词器甲言比较后，F 值提高了 14%，取得了良好的效果，证明基于大规模古文语料库建立的古文词典，能够提供良好的古文分词效果。

关键词：古汉语分词；大数据；语料库

中图分类号：TP391 **文献标识码**：A

Large-scale Online Corpus Based Classical Integrated Chinese Dictionary Construction and Word Segmentation

XING Fugui[1,2], ZHU Tingshao[1,2]

(1. Institute of Psychology, Chinese Academy of Sciences, Beijing 100101, China;
2. Department of Psychology, University of Chinese Academy of Sciences, Beijing 100049, China)

Abstract: The classical Chinese word segmentation is an important step to analyze existing ancient documents. In this paper, we first collect unstructured classical Chinese online corpus and accumate a basic dictionary. Then the candidate new words are discovered by a multi-feature fusion strategy, including mutual information, information entropy, and position word probability. Finally, a CCIDict of 349,740 words is applied with the forward maximum matching to segment the words in classical Chinese texts, achieving 14% improvements in F-value compared with the open-source Jiayan.

Keywords: classical Chinese word segmentation; big data; corpus

0 引言

针对中国古代文献的挖掘与分析，有助于传统文化的继承与弘扬[1]，而古文分词是古代文献分析的重要环节。由于古文和现代汉语无论在词汇还是语法上都有很大的不同，探索古文的有效分词是亟待解决的问题。

相关学者在古文分词方面做了大量研究。严顺[2]提出了基于条件随机场(CRF)的古汉语分词模型；王晓玉等[3]将 CRFs 模型和词典相结合，用于古汉语的分词；钱智勇等[4]使用隐马尔可夫模型对《楚辞》进行了自动分词标注的研究实验，取得了较好的效果；李筱瑜[5]将互信息、信息熵相结合进行新词发现，并与词典信息结合用于古籍文本分词研究，但是准确度有待提高。

目前中文分词算法[6]有很多，根据其工作原理分为三种：基于词典、基于统计及基于理解的分词。其中，基于词典的分词算法速度最快，因此被广泛使用。基于词典的分词算法在匹配顺序上分为正向、逆向和双向，而在匹配方法上分为最大、最小和最佳。

收稿日期：2020-04-16 **定稿日期**：2020-07-07

基金项目：国家社会科学基金(17AZD041)

基于词典的分词算法的核心是词典的构建，本文提出利用大规模在线古文语料库，通过分析海量古文资料[7]，进行古文词典的构建。通过网络爬虫实现海量在线古汉语数据集的获取，经过数据清洗和转换得到古汉语基础词典。这个基础词典虽然是经过对大规模在线非结构化数据转换所得，囊括了大部分古汉语字词，但其包含的古汉语词汇的全面性仍有不足。为此，本文提出互信息、信息熵[8]和位置成词概率[9]相结合的古文新词发现方法，从大规模古籍文本中抽取古汉语词汇并得到候补词典，以此来弥补基础词典在全面性方面的不足。通过融合基础词典与候补词典，得到最终的集成古文词典(classical Chinese integrated dictionary，CCIDict)。基于CCIDict，利用不同的分词方法实现对古文的分词，并与开源古汉语分词器甲言①进行性能的比较，检验了不同分词算法在CCIDict上的表现。

1 研究方法

1.1 基础词典的构建

在互联网发达的今天，各种在线古汉语数据资料日渐丰富，包括GitHub和百度上收录的大量开放的古汉语数据集和资料。除此之外，汉文学网②、词典网③、汉典④、在线汉语字典⑤、国学大师⑥等词典网也收录了大量古汉语字词。其中，汉文学网收录了新华字典、汉语词典、成语词典和文言文字典；在线汉语字典收录了大量现代和古代字词，每个字包含读音、解释、部首、笔画和拼音，使用者根据字可以查到相关的词语，但是字和词并没有进行古今区分；词典网收录了汉语词典35万多条；汉典收录了超过8万个汉字、20万个词语；国学大师收录了汉语词条74万。本文根据网站的robots协议和许可限制开发了爬虫系统，下载得到各个网站的非结构化古文字词数据，并利用Hadoop文件系统对收集到的大规模古文资料进行管理和维护。

本文经过对原始数据的整理、清洗和转换，最终抽取了22 203个字、364 761个词语。但是这些字和词语还包括一些现代字词，因此需要对数据做进一步处理，以去除那些现代字词。为了判断某一个字/词是否为古汉语字词，我们使用了关键词搜索的方法将这些字和词作为关键词在古汉语语料库中进行检索。如果字词在语料库中存在，则认定其为古汉语字或者古汉语词语，否则不是。

在GitHub上有很多的开源古汉语语料库，汉语古典文本资料库⑦有13 000种文本、10万卷、近13亿字，大小为3.14 GB，基本上涵盖了各个朝代的古籍文献，本文选定该数据集作为筛选古汉语字词的依据。为了达到良好的搜索性能，本文使用了分布式索引框架Apache Elasticsearch[10]，并将3.14 GB的古汉语语料库建立索引，然后通过Elasticsearch提供的检索API对词典中的字和词进行筛选。如果某个字或者词语能在Elasticsearch索引库中检索到，就将这个词标记为文言词汇，最终形成一个只有古汉语字词的基础词表。本文使用IK分析插件⑧作为Elasticsearch中索引和检索的分词组件，IK的优点是能够精确地按照中文词典分词，然而IK中提供的是现代分词字典，在实验中需要进行修改，我们将上文得到的字和词转换成字典文件，替换了IK中原有的字典文件。

Elasticsearch有三种搜索方式：term、match和match_phrase，其中match和match_phrase都会将搜索关键词拆分成子关键词，然后对子关键词进行二次检索，所以该方法无法满足本实验的要求，term搜索方式能够做到整词匹配，虽然会对字典中不存在的词语进行单字拆分，但是由于我们只针对字典中的词进行检测，所以字典之外的词语不会被作为关键词进行搜索。

经过去重处理和简繁体转换，我们得到了包含331 516个字词的词典，本文称之为基础词典。

1.2 新词发现

新词发现是自然语言处理中的一项重要技术，用于抽取字典中没有的新词[11]。本文在获取古文基础词典后，仍然难以确保基础词典收录了全面的古汉语字词。为了弥补基础词典的不足，在这一步骤中，我们提出了多特征融合的方法实现新词发现，综合采用N-Gram词频、互信息、信息熵、位置成词概率相结合的方式在大规模语料库中抽取新词，语料库采用上文提到的开源的3.14 GB的汉语古典文本资料库。在既往研究中也有采用新词发现的方式

① https://github.com/jiaeyan/Jiayan/
② https://www.hwxnet.com/
③ https://www.cidianwang.com/
④ https://www.zdic.net/
⑤ http://xh.5156edu.com/
⑥ http://www.guoxuedashi.com/
⑦ https://github.com/mahavivo/scripta-sinica
⑧ https://github.com/medcl/elasticsearch-analysis-ik

抽取古汉语词汇[5]，但是这些方法均在少量的语料库上进行实验，效果并不理想，而在本实验中采用大规模语料，并使用分布式计算进行新词发现。计算如式(1)、式(2)所示。

$$C_s(u)=\bigwedge_{\text{gram}^T}^{F(w)} f_{\text{cluster}}(s) \tag{1}$$

$$F(w)=\text{PMI}(w)\,\text{En}(w)\,\text{PWP}\,(w,\text{pos})_{\text{Bdict}} \tag{2}$$

其中，$s=[s_1,\cdots,s_n]$ 代表由多个文本文件构成的大规模语料，$f_{\text{cluster}}(s)$ 代表通过分布式计算对大规模语料进行处理的新词发现函数，$C_s(u)$ 代表基于大规模语料 s 通过新词发现得到的古汉语候补词语集合，u 表示最佳参数集，$F(w)$ 代表成词的多特征，$F(w)$ 由互信息 $\text{PMI}(w)$、信息熵 $\text{En}(w)$、位置成词概率 $\text{PWP}\,(w,\text{pos})_{\text{Bdict}}$ 共同决定，w 表示候选词，$\text{PWP}\,(w,\text{pos})_{\text{Bdict}}$ 根据基础词典计算所得，Bdict 表示基础词典，pos 代表位置，分为词首、词中和词尾。gram 代表 N-Gram 中的单元，N-Gram 是一个统计语言模型[12]，其假设第 N 个词的出现只与前面 N-1 个词相关，而与其他的词都不相关。

本文首先对古汉语文本进行切分，切分后的每个单元被认为是一个 gram，进而统计出每个 gram 的频率。根据设定的阈值，过滤掉不符合词长要求的词语。由于 N-Gram 的语言无关性，无须对古汉语文本进行语言学处理。Trie 树是一个树形结构，是哈希树的变种[13]。Trie 树利用前缀来缩短检索时间，能够最大限度地减少字符串的比较，尤其在动态地增加或者修改数据的场景下性能表现更好，本文在此基础上构建了 gram^T 词频树。

互信息和信息熵都是信息论中的概念，互信息可以计算两个对象的关联程度，如果 X 和 Y 互相独立，那么 X 和 Y 之间互相不提供任何信息，它们的互信息就为 0。N-Gram 获取了高频率的文本片段，但是一个文本片段出现的频率高并不能代表这个文本片段是一个真正的词语，它可能是多个词语结合在一起的词组。本文采用互信息度量不同文本片段的凝固程度。互信息 $\text{PMI}(w)$ 的计算如式(3)、式(4)所示。

$$\text{PMI}(w)=\log_2\frac{p(w)}{\text{avg}(w)} \tag{3}$$

$$p(w)=\frac{f(w)}{\text{num}} \tag{4}$$

其中，$w=[w_1,\cdots,w_n]$ 代表由多个字构成的词语，$p(w)$ 代表词语 w 在大规模语料库中出现的概率，$f(w)$ 代表词语 w 在大规模语料中出现的次数，num 表示大规模语料库的字数，$\text{avg}(w)$ 表示词语中的字不同组合的平均概率。

信息熵也被称为自由度，用于判断古汉语文本左右相邻字符的相互关系的稳定性，熵越大，信息量就越大，不确定性也越高，由此反映左右相邻字符的搭配是否丰富。信息熵分为左信息熵和右信息熵。左信息熵代表一个文本片段与左边的字符相结合的稳定程度，右信息熵代表一个文本片段与右边的字符相结合的稳定程度，计算如式(5)所示。

$$\text{En}(w)=-\sum_{w_{\text{left}|\text{right}}} p(w_{\text{left}|\text{right}})\log_2 p(w_{\text{left}|\text{right}}) \tag{5}$$

其中，$p(w_{\text{left}|\text{right}})$ 是出现候选词 w 时，其左边或者右边相邻字符 $w_{\text{left}|\text{right}}$ 的条件概率。$w_{\text{left}|\text{right}}$ 的计算如式(6)所示。

$$p(w_{\text{left}|\text{right}})=\frac{N(w_{\text{left}|\text{right}})}{N(w)} \tag{6}$$

其中，$N(w_{\text{left}|\text{right}})$ 是相邻字符及候选词 w 共同出现的概率，$N(w)$ 是候选词 w 单独出现的概率。

通过词频统计并搭配互信息和信息熵得到的词典已经涵盖了足够多的候补词语，但是仍然存在一些非古汉语词语。这些非古汉语词语有些是完整词语的部分片段，例如，“氏家世”并不是一个完整的词语，一般用法为“刘氏家世”“家世”等。为了进一步提高成词的准确度，本文通过位置成词概率对上一步得到的词典进行过滤。在古汉语中，每个字或者词语都有自己的构词规律，某个字会出现在合成词的固定的位置，例如“才”一般出现在某个词的结尾，如“秀才”“王进才”等。所以，在结尾这个位置的概率比较高。位置成词概率的计算如式(7)所示。

$$\text{PWP}\,(w,\text{pos})_{\text{Bdict}}=\frac{N(w,\text{pos})}{N(w)} \tag{7}$$

其中，pos 表示古汉字 w 在该词中出现的位置，位置包括词首、词中、词尾，$N(w,\text{pos})$ 表示古汉字 w 出现的位置 pos 在所有词语中的频次；$N(w)$ 则表示 w 在基础词典 Bdict 的词语中出现的总次数。假设一个词语词首含有某个古汉字，这个古汉字在词首的位置的概率越小，表明这个词语成为古汉语词语的可能性越小；同理，如果一个古汉字在词尾的位置的概率越小，表明这个词语成为古汉语词语的可能性也越小。

由于新词发现需要在整个语料库上进行计算，只有在整个语料库上重复出现过的字符串才可能是候选词，因此语料库越大，时间复杂度也越大。本文基于 Spark 实现了并行处理[14]，通过 Spark RDD 实

现了分布式的 Trie 树，在效率上有了提升[15]。RDD 是弹性分布式数据集[16]，是 Spark 实现分布式处理的基石，它将数据分布在多台机器上，提高了计算性能。算法基本流程如算法 1 所示。

算法 1　新词发现算法

输入：基础词典 **Bdict**，原始语料 $\boldsymbol{s}=[s_1,\cdots,s_n]$，最小词长 min_***wl***，最大词长 max_***wl***，最小词频 min_***wf***，最小互信息阈值 min_***pmi***，最小信息熵阈值 min_***en***。

输出：候补词语集合 $\boldsymbol{C}_s(u)$。

Begin

① 基于基础词典 Bdict 根据式(7) 计算位置概率。

② 以 RDD 形式加载原始语料 s 到内存，从整个语料根据最大词长 max_wl 按照字构建 Trie 树并计算 gram 词频，使用 broadcast 操作将建好的 Trie 树广播到其他计算节点，减少计算中的通信开销。计算得到词频树 gram^{T}。

Repeat 遍历 gram^{T}

③ 根据式(3)、式(4)计算互信息。

④ 根据式(5)、式(6)计算左信息熵和右信息熵。

Until 计算完成所有 gram 单元

⑤ 根据式(1)、式(2)按照最小词长 min_wl，最大词长 max_wl，最小词频 min_wf，最小互信息阈值 min_pmi，最小信息熵阈值 min_en 过滤出符合条件的词语，生成 $C_s(u)$。

Return $C_s(u)$

End

本文基于算法 1 进行实验，得到候补词典。计算节点配置为，CPU 8 Core，Intel Xeon；内存 64 GB；操作系统为 Ubuntu 18.04.3 LTS；JDK 版本为 1.8；Hadoop 版本为 3.2.0；Spark 版本为 2.4.4。根据启发式参数调优策略，本文的参数配置如下：最小词频为 10、最小词长为 2、最大词长为 8、最小互信息阈值为 0.2、最小信息熵阈值为 0.2、最小词首概率为 0.4、最小词尾概率为 0.6。

基于式(7)得到位置成词概率表并绘制成散点图，如图 1 所示。

从图 1 可以看出，词中位置概率主要集中在 0.2 以下，在提高成词准确度方面并不明显，所以本文只考虑词首概率和词尾概率。

1.3　基于 CCIDict 的古文分词系统的构建

我们将基础词典和候补词典整合在一起，通过冗余处理合并了相同的词语，形成集成古文词典(CCIDict)，词典分成两列，第一列为字词，第二列为词频。

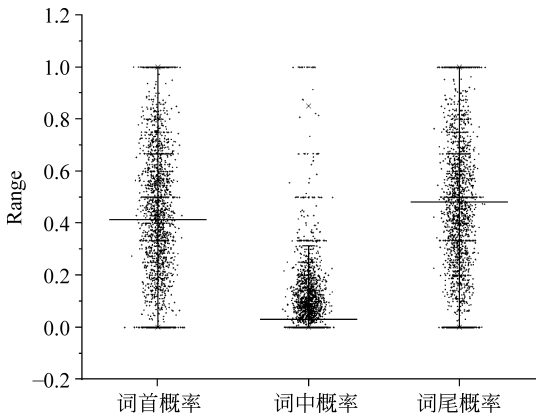

图 1　位置概率散点图

2　结果与分析

2.1　测试数据集

为了测试古文分词的准确性，本文使用人工标注的语料库作为测试依据。语言数据协会①的汉语语料库(LDC2017T14)包含了《左传》的分词和词性标注，共 18 万个中文字符，由两部分构成，训练数据(166 138 字)和测试数据(28 131 字)，然而该数据只包含《左传》的数据，数据的多样性略显不足。

为此，我们对标注数据进行了扩展，增加了不同朝代具有代表性的文言语料，包括春秋、战国、秦朝、汉朝、魏晋南北朝、隋唐、宋朝、辽金元、明朝及清朝。然后，组织古汉语专业研究生作为被试，对这些新增语料进行人工标注，最终与《左传》的数据集进行合并，作为测试集。在人工标注实验中，我们根据不同朝代选取了共 40 篇文章，按照文本量大小分成 6 部分，发放给 6 名被试进行人工标注。被试根据指导语按照自己的理解，使用空格标记出古汉语词和字，选取的古籍文本出处如表 1 所示。

表 1　人工标注文本的出处

朝代	节选出处
春秋	《史记·田敬仲完世家》《史记·晋文称霸》《论语·学而篇》《邓析子·无厚》《史记·项羽本纪》
战国	《史记·孟尝君列传》《全上古·上书说秦昭王》《楚辞·离骚》《孟子·梁惠王》《孟子·公孙丑》
秦朝	《史记·秦始皇本纪》《韩非子·存韩·上书韩王》《史记·李斯传·上书对二世》

① https://catalog.ldc.upenn.edu/LDC2017T14

续表

朝代	节选出处
汉朝	《论衡》《汉书·董仲舒传》《贾谊传》《史记·扁鹊仓公列传》
魏晋南北朝	《傅子》《抱朴子》《三国志·魏书·武文世王公传》《陆景》《典语》
隋唐	《史通·自叙》《长乐老自叙》《与文徵明书》《先侍御史府君神道表》
宋朝	《金石录后序》《指南录后序》《训俭示康》《九议》
辽金元	《辽史》《归潜堂记》《金史》《元史·本纪第一》
明朝	《御制皇陵碑》《前历试卷自序》《白牛生传》《立命之学》
清朝	《三十自述》《三侬赘人广自序》《弢园老民自传》《与弟文韶书》

2.2　词典扩展前后的分词效果比较

由于古汉语和现代汉语在验证准确度的方法上是一样的，所以本文使用了 Bakeoff 2005 数据集[①]包含的 Perl 脚本，并采用准确率 P(Precision)、召回率 R(Recall)和 F 值(F-measure)作为分词的评价指标。我们将扩展的测试集作为标准切分语料，并使用 Bakeoff 提供的脚本对分词结果进行评估。本实验选用正向最大匹配作为分词算法，分别基于基础词典和扩展后的 CCIDict 词典进行分词，性能比较如图 2 所示。

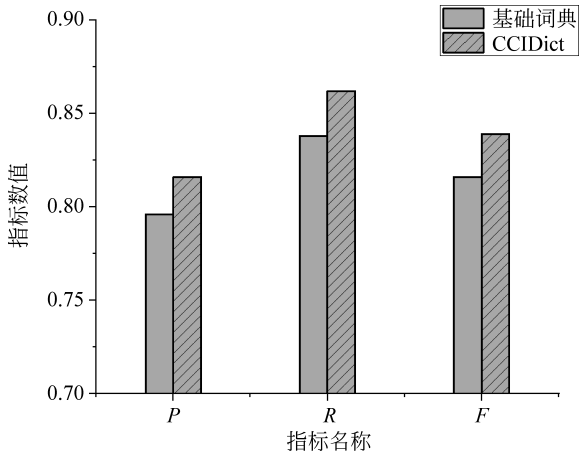

图 2　词典扩展前后性能对比图

从图 2 的比较可以看出，将基础词典和候补词典合并之后，分词的准确性得到了提升。

2.3　分词性能比较

为了评估分词效果，本文将基于 CCIDict 的分词器与甲言分词器进行比较。甲言分词器中包含了两种分词模式：基于词典的分词和基于 HMM 模型的分词。我们分别与这两种模式的分词进行了比较。甲言分词器虽然有新词发现功能，但是单机程序时间复杂度太高，不适合在大规模语料上进行新词发现，所以在本实验中甲言分词器使用自带的词库。

我们将待切分的文本分别输入两个分词器进行分词，并输出各自的评价指标，包括准确率 P、召回率 R 和 F 值。基于词典的分词算法可以细分为多个小类，为了探索出最佳的分词性能，本文选取了 12 种算法组合。其中最大匹配和最小匹配指的都是词长的匹配，根据 CCIDict 词语特点将正向和反向最大匹配的最大词长设置为 9；双向匹配根据总词数、单字词数和非字典词数计算出正向匹配和反向匹配的惩罚分数，将惩罚分数最小的一个作为最终分词结果；最大概率匹配指的是基于动态规划算法，匹配句子概率路径中概率最大的词；最大词频匹配指的是匹配词频最大的词语，词频为大规模语料库中该词的词频。分词结果如表 2 所示。

表 2　分词性能比较

分词条件		P	R	F
甲言词典分词		0.651	0.751	0.698
甲言 HMM 分词		0.750	0.798	0.773
CCIDict	正向最大匹配	0.816	0.862	0.839
	正向最小匹配	0.619	0.776	0.689
	正向最大概率匹配	0.815	0.860	0.837
	正向最大词频匹配	0.620	0.777	0.690
	反向最大匹配	0.807	0.852	0.829
	反向最小匹配	0.617	0.774	0.687
	反向最大概率匹配	0.618	0.774	0.687
	反向最大词频匹配	0.618	0.774	0.687
	双向最大匹配	0.807	0.852	0.829
	双向最小匹配	0.617	0.774	0.687
	双向最大概率匹配	0.618	0.774	0.687
	双向最大词频匹配	0.618	0.774	0.687

结果显示，基于 CCIDict 和正向最大匹配算法构建的分词器与甲言相比，在准确率、召回率及 F 值这些指标上均有明显的提升，CCIDict 古文分词

① http://sighan.cs.uchicago.edu/bakeoff2005/

系统的 F 值比甲言的词典分词模式提高了近 14%，取得了更好的效果。由此表明，基于互联网大规模语料库的词典构建在古汉语分词上是可行的，并且是有效的。

3 总结

中国古代文献是一个宝藏，从古汉语文本中挖掘有价值的信息在考古中有很重要的意义。本文在整合在线古汉语数据资源的基础上，生成基础词典。利用 N-Gram、互信息、信息熵、位置成词概率等多特征融合的新词发现方法，在大规模语料库上抽取古汉语词汇，形成 CCIDict。在 CCIDict 的基础上，利用多种分词算法实现古文的分词，通过与甲言进行比较，基于 CCIDict 的正向最大匹配分词算法在测试集上取得了良好的效果。结果表明，本文提出的方法是可行的，分词准确率有了较大的提高。

本研究也存在一定的局限。目前采集的古汉语语料数据源较少，在以后的研究中将进一步扩大数据来源，以收集更全面的古汉语数据；在新词发现中的超参数的设置还有待进一步优化；本文目前没有对歧义词进行处理，在未来工作中，将借鉴 AdaBoost.MH 算法[17] 在歧义词处理方面的思想，实现古文分词中的有效排歧。

古文的有效分词是针对古文献挖掘的重要环节，针对古文献的研究有助于我们深入了解中国文化的变迁，为进一步弘扬传统文化提供技术支撑。

参考文献

[1] Amrani A, Abajian V, Kodratoff Y, et al. A chain of text-mining to extract information in archaeology[C]// Proceedings of the 3rd International Conference on Information and Communication Technologies: From Theory to Applications, 2008:1-5.

[2] 严顺. 基于 CRF 的古汉语分词标注模型研究[J]. 江苏科技信息,2016,(8):14-16.

[3] 王晓玉，李斌. 基于 CRFs 和词典信息的中古汉语自动分词[J]. 数据分析与知识发现，2017，1(5):62-70.

[4] 钱智勇，周建忠，童国平，等. 基于 HMM 的楚辞自动分词标注研究[J]. 图书情报工作,2014,58(4):105-110.

[5] 李筱瑜. 基于新词发现与词典信息的古籍文本分词研究[J]. 软件导刊,2019,18(4):66-69.

[6] 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.

[7] 刘永楠，李建中，高宏. 海量不完整数据的核心数据选择问题的研究[J]. 计算机学报,2018,40(4):915-930.

[8] 夭荣朋，许国艳，宋健. 基于改进互信息和邻接熵的微博新词发现方法[J]. 计算机应用，2016,36(10):2772-2776.

[9] 林自芳，蒋秀凤. 基于改进位置成词概率的新词识别[J]. 福州大学学报(自然科学版),2011,39(1):43-48.

[10] Voit A, Stankus A, Magomedov S, et al. Big data processing for full-text search and visualization with elasticsearch[J]. International Journal of Advanced Computer Science and Applications, 2017, 8(12):76-83.

[11] 王思丽，祝忠明，刘巍，等.领域本体学习语料的自动获取与预处理方法研究[J].图书馆学研究，2019，(20):54-64.

[12] 鲁一冰，刘驰. Skip-ngram 模型解决数据稀疏问题的研究[J]. 自动化技术与应用,2015,34(3):35-37,46.

[13] 王思力，张华平，王斌. 双数组 Trie 树算法优化及其应用研究[J]. 中文信息学报,2006,20(5):26-32.

[14] Hou J, Zhu Y, Du S, et al. Design and implementation of reconfigurable acceleration for in-memory distributed big data computing[J]. Future Generation Computer Systems, 2019, 92(3): 68-75.

[15] 刘鹏，滕家雨，丁恩杰，等. 基于 Spark 的大规模文本 k-means 并行聚类算法[J]. 中文信息学报,2017,31(4):145-153.

[16] Aziz K, Zaidouni D, Bellafkih M. Big data optimisation among RDDs persistence in apache spark[M]. Communications in Computer and Information Science, 2018: 29-40.

[17] 刘风成，黄德根，姜鹏. 基于 AdaBoost.MH 算法的汉语多义词消歧[J]. 中文信息学报,2006,20(3):8-15.

邢付贵(1987—)，硕士研究生，主要研究领域为大数据技术、机器学习、自然语言处理、应用心理学。

E-mail: 446513370@qq.com

朱廷劭(1971—)，通信作者，博士，主要研究领域为机器学习、应用心理学、大数据技术。

E-mail: tszhu@psych.ac.cn