情报工程 ISSN: 2095-915X
第1卷 第2期 64-72 2015年4月
TECHNOLOGY INTELLIGENCE ENGINEERING
ISSN: 2095-915X Vol.1 No.2 64-72 Apr 2015

doi:10.3772/j.issn.2095-915x.2015.02.008

基于最大熵模型的学术缩写自动识别 *

张秋子，陆伟，程齐凯，黄永

（武汉大学信息资源研究中心 湖北武汉 430072）

摘要：为实现海量英文学术文本中缩写词及对应缩写定义的识别，本文提出了一种自动缩写识别算法MELearn-AI。该算法在人工标注数据集的基础上，从序列标注的角度，通过最大熵模型实现了计算机领域英文学术文本中的自动缩写识别。MELearn-AI在本文构建的评测数据集“Paren-sen”上得到了95.8%的查准率和86.3%的查全率，相对于其他两组对照实验的效果有较为明显的提升。本文提出的自动缩写识别方法能够在计算机领域的学术文本上取得令人满意的效果，有助于更好地理解并利用该领域术语。

关键词：学术文本，缩写，机器学习，序列标注，信息抽取

中图分类号：G203

Study on Automatic Identification of Academic Abbreviations and their Definitions based on Maximum Entropy Model

ZHANG Qiuzi, LU Wei, CHENG Qikai, HUANG Yong

(Center for the Studies of Information Resources of Wuhan University, Wuhan 430072, China)

Abstract: In order to effectively identify the abbreviations and their corresponding definitions from enormous English academic texts, this paper proposes an automatic identification algorithm called MELearn-AI.In the perspective of the sequence labelling,MELearn-AI utilizes a manually labelled dataset and adopts maximum entropy algorithm to train a model, and then identify abbreviations in computer science academic texts based on the model. This method achieves a 95.8% precision rate with a 86.3% recall rate in the "Paren-sen" evaluation dataset created in this paper,it shows an obvious improvement compared to the other two algorithms.This paper proposes a method to identify the abbreviations and their corresponding definitions.Tested in English academic texts of computer science, the algorithm achieves satisfactory results, which is helpful to better understanding and adopting the terminology of this field.

Keywords: Academic texts, abbreviations/acronyms, machine learning sequence ,labelling, information extraction

基金项目：本项目得到国家自然科学基金，“基于语言模型的通用实体检索建模及框架实现研究”（项目编号：71173164）支持。
作者简介：张秋子（1992-），女，硕士研究生，主要从事信息检索与数据挖掘方向的研究；陆伟（1974-），男，博士，教授，主要从事信息检索、知识管理、知识挖掘方向的研究；程齐凯（1989-），男，博士研究生，主要从事信息检索和数据挖掘方向的研究；黄永（1991-），男，博士研究生，主要从事信息检索与数据挖掘方向的研究。

1 引言

缩写是现代英语中一种普遍存在的构词现象，缩写的类型丰富多样，包括长单词的缩略、名词词组的首字母缩写、专业术语的缩写等等。缩写一方面带来了行文和交流的简便，但另一方面也对用户理解文献带来了一定障碍。在科研文献，甚至于一般的文献中，给定一个缩写，如SVM、Maxent，用户需要一定的背景知识才能理解特定的缩写词汇。为了更好的帮助用户理解相关概念，缩写识别成为了科研文献内容挖掘的一个重要问题。

缩写识别中的两个重要概念是缩写词和缩写定义，缩写词（abbreviation, or acronym）指用来代替原有词或词组的简洁的表达方式，缩写定义（abbreviation definition）指缩写词在被缩写之前较长并且相对完整的表达形式。缩写识别包括缩写词的识别以及缩写定义的识别两个部分。缩写词的识别要求辨别出学术文本中哪些是缩写词，缩写定义的识别要求正确地抽取出该缩写词对应的缩写定义。例如：

例 1: the methodis Support Vector Machine (SVM) ——><Support Vector Machine，SVM> 正确识别，缩写词是 SVM，缩写定义是 Support Vector Machine。

例 2 :such as the undrained shear strength (Su) ——><shear strength，Su> 错误识别，Su 被误认为是缩写词。

缩写自动识别具有重要的应用价值：能够用于辅助构建领域相关数据集 [1] 或编制领域词典 [2]；也是自然语言处理和信息抽取工作的重要基础部分 [3]；还可以用于查询消歧以优化搜索引擎尤其是学术搜索引擎的性能 [3]；同时，缩写识别还在知识发现、本体构建、自动问答等知识层面的研究领域具有潜在应用价值。

目前国内外关于缩写识别的研究大都集中于缩写较为规范的生物医学领域 [2, 4–9]，针对计算机科学领域的研究较少。但是计算机科学领域的缩写词增长和更新速度快 [10]，一词多义情况严重，研究该领域的缩写识别问题也具有上述相关的研究价值和应用价值。本文提出了一种基于最大熵模型（Maximum Entropy Model）的机器学习识别方法——MELearn–AI, 以期解决计算机领域的缩写自动识别问题。

本文的内容组织如下：第二部分出了缩写识别的研究综述；第三部分确定缩写抽取范围，并详细论述 MELearn–AI 算法的思路；第四部分介绍本文使用的评测数据集和评价指标，进行对比实验并分析实验结果；第五部分总结本文工作的成果和局限，并给出进一步工作的研究方向。

2 相关研究

缩写识别的目标任务是找出缩写词（short form, 即 SF）及其对应的缩写定义（long form，即 LF），即找出 <“short form”，“long form”> 这样的组对。目前主流的抽取算法分为两类：第一类是基于规则的方法，第二类是基于统计与学习的方法。

基于规则的方法是最常用的方法，其中缩写词的位置大都通过小括号作为标志进行确定 [1, 3, 4, 9, 11, 12]，少数方法考虑了其他的形式，比如加上对中括号的判别 [7]，加上文本标记和线索词等其他情况 [2, 5]，或利用某些语法信息 [13] 辅助确定。缩写定义的抽取方法基本上可分为两类：一是通过一系列启发性的模式匹配规则进行获取 [2, 4–6, 14]；二是先利用一定规则得到候选定义集合，再通过比对算法计算 LF 和 SF 之间关联性或是匹配度从而选出最优定义 [7–9, 12]。不同于上述两类方法，Sohn 和 Comeau 等人 [1] 创新性地采用了一种伪准确率算法，能够预先计算一个缩写词采用不同策略抽取出的缩写定义的可靠性，从而在抽取过程

中选择可靠性最高的策略。

基于统计与学习的方法又可分为两类，一是单纯利用数据集中有关缩写的统计信息进行识别，二是借助各类机器学习模型进行学习。

利用的统计信息可以是搭配频次，比如 Zhou 等人[11]通过分析 MEDLINE 中型为“LF(SF)”的搭配；或者是统计学指标，比如 Hisamitsu 和 Niwa[15] 利用的互信息、卡方检验等指标；也可以是词项频次，比如 Okazaki 和 Ananiadou[16] 先统计在文本中出现较为频繁的词项，然后计算这些词项成为缩写定义的可能性。

Chang 等人[12]利用有监督的机器学习方法帮助识别缩写。他们通过 LCS 算法找出所有可能的缩写词和定义对，使用二元逻辑回归对所有可能性进行分类，最终他们在 Medstract[1] 数据集中得到 80% 的准确率和 83% 的召回率。Nadeau 和 Turney[17] 同样利用有监督的机器学习方法进行抽取，但是他们选择用较为宽松的限制来增加召回率。Dannells[18] 则采用基于记忆的学习算法（Memory Based Learning，MBL）。

作为本文的对比算法之一，Liu 和 Friedman[19] 提出了一种挖掘小括号内外搭配情况的方法。他们抽取出在文本中出现多于一次的搭配并且通过 3 项规则来剔除不可能的候选项。他们在自己人工标注创建的评价数据集上得到了 96.3% 的准确率和 88.5% 的召回率，但是不能抽取在数据集中只出现过一次的缩写定义。

另一种对比算法是 Sanchez 和 Isern[20] 提出的 WEB 网页自动缩写识别算法，依次按照缩写词集合的生成，缩写定义检索和定义可靠性评价与过滤的步骤进行。他们要求缩写定义满足表 1 所示规则：

综上，基于规则的方法往往需要编制非常繁琐的规则才能保证较高的准确率；单纯利用统计信息进行缩写识别倾向于识别高频的缩写和缩写模式，对于长尾情形的识别效果较差；而借助机器学习模型进行识别具有更强的文本内容适应性和领域拓展性，并且不存在对低频缩写识别不足的问题，但是对训练数据和训练模型的依赖性强。

为了更好地发挥机器学习方法的优势，并有效规避上述缺陷，本文选择在句法分析、信息抽取等多个自然语言处理问题上得到广泛应用的最大熵模型作为分类模型，通过人工标注的方式构建训练集，将缩写识别问题转化为词汇的分类问题。

表 1 Sanchez 和 Isern 关于缩写定义的限制

规则	规则描述
规则 1	All acronym characters must appear in the definition
规则 2	Acronym characters must appear in the same order asin the definition
规则 3	Definition must begin with the same letter as theacronym
规则 4	Definition maximum length is n10, where n is thenumber of acronym characters
规则 5	Definition must have at least one more character thanthe acronym

1 http://www.medstract.org/

3 研究方法

本文的目的是抽取计算机领域英文学术文本中的缩写，首先需要确定缩写抽取的范围，其次是选择恰当的分类模型，以及分类特征。故本节将从上述三个方面予以论述。

3.1 缩写抽取范围

（1）缩写出现形式

英语文本中的缩写词多数出现在小括号内部，即 definition(abbreviation) 的形式，同时伴随少数 abbreviation (definition) 的形式。Wren 在文献中人工检查了 100 篇摘要，在出现的 169 个缩写词中只有 4 个不以小括号形式出现，并且所有以小括号的形式出现的缩写都是 definition(abbreviation) 的情况 [3]。同样的，笔者以 support vector machine (SVM) 为例，抽取出 2323 句同时含有“SVM”以及 support vector machine/support vector machines/support−vector−machine/support−vector−machines 四个词组之一的句子，人工检查他们的共现情况，只有极少数情况下“SVM”不在小括号内出现，而是用“for short”，“stand for ”等词组连接。综合 Wren[3] 和笔者的实证调查情况，本文做出以下假设：缩写词在文章中第一次出现的时候，会伴随其定义一起出现，并且缩写词在小括号内部，其定义邻近小括号在其左侧。

（2）缩写类型

本文以计算机科学领域中的缩写为研究对象，这些缩写往往以术语形式存在。对于其他领域的缩写词，如数学论文中常出现的 infimum(*inf，即下确界）和 supremum(*sup，即上确界)[21] 等缩写情形，并不在本文研究范围之内；而对于 exempli gratia（e.g.）等领域独立的缩写，本文亦不予以考虑。

本研究关注的缩写可分为三类：一是首字母缩写词，即由词组中核心词的首字母组合而成 [22, 23]，该类在缩写中最为普遍，如 Support Vector Machine（SVM）；二是部分缩短词，即对原来的词进行加工，截去其中一部分字母或音节构成新词 [24]，如 acknowledgements(ACKs)，electroencephalogram (EEG) 等；最后是以阿拉伯数字代替英文数字，再和量词的首字母组合而成 [25] 的特例情况，如 three dimensional(3D)。

3.2 分类模型

香农认为信息是人们对事物了解的不确定性的消除或减少，他把不确定的程度称为信息熵 [22]。最大熵模型的基本思想是为所有已知的因素建立模型，而把所有未知的因素排除在外 [23]。最大熵模型重要特点是它不要求特征满足条件独立，故人们可以相对任意地加入对最终分类有用的特征，而不用顾忌特征之间的相互影响。并且该模型输出的是一个相对客观的概率值结果，便于后续推理步骤使用 [14]。

本文任务中会涉及到多种特征，假设 x 是这些特征构成的向量，即特征向量，变量 y 的值为句子中各个单词的类别，即“boundary”、“continue”以及“other”。模型的训练便是从人工标注的数据集中学习上述三种类别的特征，从而在测试集中区分出上述三类。

①如果 <inner> 标签中是缩写词，并且其前方文本包含对应的缩写定义，那么将缩写定义的第一个单词标记为“boundary”，缩写定义中非

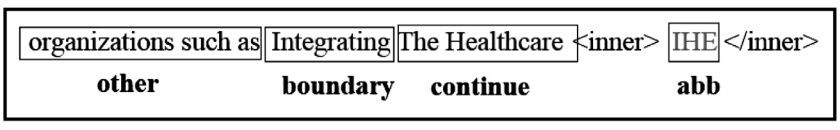

图 1 训练集标注样例 1

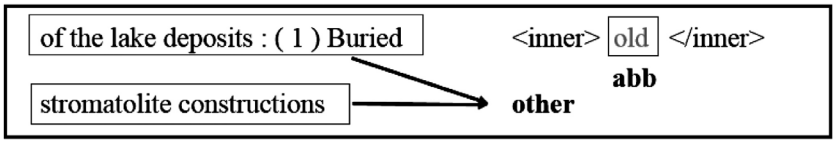

图 2 训练集标注样例 2

开头的单词标记为“continue”，该句子中的其他词标记为“other”，如图 1 所示。

②如果在 <inner> 前方文本中找不到 <inner> 标签中候选缩写词对应的定义，则将前方所有单词标记为“other”，如图 2 所示。

最大熵模型要求 p(y|x) 在满足一定的约束条件下，必须使下面定义的熵取得最大值：

$$H(p) = -\sum_{x,y} p(y|x)\log p(y|x) \quad (1)$$

这里需要引入多种特征表明 x 与 y 存在某种特定的关系，一般用以下的方式表述：

$$f_i(x,y) = \begin{cases} 1, & if(x,y) \text{ 满足某种条件} \\ 0 & \text{否则} \end{cases}, \quad i = 1, 2, 3, \ldots, n \quad (2)$$

3.3 训练特征

本文以 support vector machine (SVM) 为例，人工检查抽取出的 2323 句子，在表 2 中归纳出计算机科学领域文本中缩写的模式。从表 2 中可以看出，缩写模式十分复杂多样，通过基于规则和单纯基于统计信息的方式进行识别是相对困难的，

表 2 缩写的模式及其相应举例

模式	举例
首字母缩写词，全部匹配	support vector machine (SVM)
	support-vector-machine (SVM)
修饰词 / 限定词 + 中心词	least-square support vector machine (LS-SVM)
	1-norm support vector machine
	proximal support vector machine
中心词 + for(如用途)	hybrid robust support vector machine for gression (HRSVMR)
中心词 + with (如附加方法、算法等)	support vector machine with automatic confidence (SVMAC)
	support vector machine with binary tree architecture (SVM-BTA)
其他算法 / 方法 + with + support vector machine	genetic algorithm with support vector machine (GA-SVM)
多种缩写在一起	Knowledge based proximal support vector machines (KBPSVMs), KBSVM1 (KBSVM with L1-norm of w),KBSVM2 (KBSVM with L2-norm of w)
方法一 + and + 方法二	hybrid support vectormachines for regression and Gaussian process for regression (SVMR-GPR)
随意穿插	Cross-validation error of conventional (SLS-SVM) and improved (ISLS-SVM) sparse least-squares support vector machines

而基于机器学习的方法是可行的。

表 2 缩写的模式及其相应举例

根据表 2，选择的特征可从三个方面予以考虑：前方文本与小括号中字母的对应情况；前方文本的停用词使用情况；前方文本含有特殊符号的情况。

表 3 将用下面的文本片段作为实例进行特征的描述说明。

例：<START> this paper proposed an accurate method based onAsymptotic Sampling（AS）<END> 对于每个句子，句首和句尾分别加入“<START>”和“<END>”以表示句子的起始和结束。记缩写词（即 abb）的长度为 len，对于每个词生成一系列特征，以上例中“Asymptotic”为例进行说明。

表 3 特征列表

特征	解释（以 Asymptotic 为例）
abb 是否全是大写字母	TRUE
（该词到待识别缩写的距离 +1）/ 待识别缩写词 abb 字数	2/2 = 1
去停用词后，（该词到待识别词缩写的距离 +1）/ 待识别缩写词 abb 字数	2/2 = 1
该词首字母是否大写	TRUE
对该词后续词汇去停用词后，该词及后续词首字母是否都为大写	TRUE
该词及后续词汇首字母包含多少个 abb 不包含的字母，不考虑大小写	0
该词及后续词汇首字母包含多少个与 abb 中字母无法一一对应的字母，不考虑大小写	0
去停用词后，该词首字母是否与 abb 首字母相同	TRUE
去停用词后，该词后续词是否与 abb 的第二个大写字母相同 || 如果 abb 内无大写字母，是否同第二个字母相同	TRUE
去停用词后，该词后续第三个词是否同 abb 的第三个大写字母相同 || 如果 abb 内无大写字母，是否同第三个字母相同 || 如果 len 小于三，认为相同	TRUE
该词后续 len 个词汇首字母是否在 abb 字母集合中	TRUE
该词及后续词首字母组成的字符串同 abb 的编辑距离（大小写敏感）	0
该词及后续词首字母组成的字符串同 abb 的编辑距离（大小写不敏感）	0
去停用词后，该词及后续词首字母组成的字符串同 abb 的编辑距离（大小写敏感）	0
去停用词后，该词及后续词首字母组成的字符串同 abb 的编辑距离（大小写不敏感）	0
该词及后续词首字母组成的字符串与去停用词后该词及后续词首字母组成的字符串的 jaccard 相似度（大小写不敏感）	1
<STRAT> 到 abb 之间的所有单词是否含有数学符号（本文考虑了 72 种数学符号，比如：“≈”，“≠”，“≤”等）	FALSE

4 实验与结果分析

4.1 数据集构建

本文使用的数据集抽样自 ScienceDirect 数据库中 2000 年至 2013 年 128 本计算机科学领域期刊的 291315 篇论文全文，预处理包括：逐篇对文本进行句子切分，共得到 11870225 句含有小括号的完整句子；去掉小括号中明显表明是公式、图表说明、引文的句子；余下的 7670445 条完整句子作为最终分析对象，记作“Paren-sen”集合。

本文从“Paren-sen”集合中随机抽取 1500 个句子进行人工标注，把其中 1300 条作为训练集，余下 200 条作为测试集。

4.2 实验设计

为测试 MELearn-AI 算法的效果，本文选取了两种目前报告效果较好且常用的算法作为基准算法（baseline）进行对比实验。

（1）对比算法

第一个对比算法源自于 Liu and Friedman 于 2003 年提出的挖掘生物医学领域术语的基于规则的缩写识别算法 [19]，按照 Liu and Friedman 原始算法的设计，从数据量极大的“Paren-sen”集合中抽取出所有缩写形式的术语是不实际也是不必要的，故本文设置抽取的缩写数量为 10000 条，再从中随机选取 200 条用作人工评测。

第二个对比算法是 David Sanchez 于 2011 年提出的从 WEB 页面抽取缩写的算法 [20]，本文将数据源从 WEB 页面转换成统一的“Paren-sen”集合，对得到的结果随机抽取 200 条作为评测。

（2）评价方法

本文对实验结果的评价采用人工检查的方法，本次实验中共有 2 名检查人员，分别对结果文件进行检查，记录下认为抽取正确的条目，再进行讨论综合双方的意见，最终确定抽取正确的条目。结果的评测指标是下列三种：

① 查准率。抽取的结果中正确的缩写定义数目与所有抽取出的缩写定义的比值。公式如下：

$$\text{Precision} = \frac{correctabb - definitionpairs}{totalabb - definitionpairs} \quad (5)$$

② 查全率。抽取的结果中正确的缩写定义数目与待抽取文本集合中所有缩写定义的比值。公式如下：

$$\text{Recall} = \frac{correct\ abb - definition\ pairas}{testset\ abb - definition\ pairs} \quad (6)$$

③ F 值。查准率与查全率的综合。公式如下：

$$\text{F} - \text{Measure} = \frac{2 \times Precision \times Recall}{Precision + Recall} \quad (7)$$

4.3 结果分析

对比实验结果如表 4 所示。

对于 Liu and Friedman 的算法，笔者从 10000 条抽取出的缩写定义中随机选取了 200 条进行评测。由于该算法需要整个数据集的统计信息作为抽取基础，故随机选取的 200 条结果无法对应出其原始待抽取集合，故在此无法评测查全率，特此说明。200 条结果中正确的共有 156 条，即查准率为 78%。抽取错误的情况主要分为两种类

表 4　实验结果对比

方法名称	Precision（%）	Recall（%）	F-Measure（%）
Liu and Friedman	78.0	--	--
David Sanchez	86.5	86.1	86.3
MELearn-AI	95.8	86.3	90.8

型：一是缩写定义前多出了充当谓语的动词，比如 MMSE 的抽取结果“estimate minimum mean square error”多出了动词“estimate”；二是缩写定义不完整，比如 MSVD 的抽取结果为“singular valudecomposit”。

对于 David Sanchez 算法，随机抽取的 200 条评测数据集中正确的缩写抽取为 173 条，故查准率为 173/200=86.5%。而 200 条缩写来自于 335 条句子（既有缩写也有非缩写），其中含有缩写句 201 条，为故查全率为 173/201=86.1%。

对于 MELearn-AI 算法，在 200 条测试集中，含有缩写定义的句子是 131 个，实验抽取出的 118 条缩写中正确的有 113 条，故查准率为 113/118=95.8%，查全率为 113/131=86.3%。经笔者对 1300 条训练集人工检查，发现含有缩写定义的句子 600 个，不含缩写定义的句子 700 个，其中不含缩写定义的句子的学习样式丰富，相对而言，含有缩写定义的句子的形式有限，故会对查全率有一定的影响。但是正是因为机器学习方法对训练数据的依赖性强，故在现有的基础上，通过增添修改训练特征或是优化训练集可进一步实现 MELearn-AI 的性能提升。总之，MELearn-AI 在训练数据有限的情况下已经较好地展示出了其相对于其他算法的优越性，并且 MELearn-AI 的扩展性和优化空间强于一般的模式匹配算法。

5 结语

缩写自动识别是信息抽取工作的一个重要组成部分，目前对于英语学术文本中缩写词及其对应定义的抽取算法主要有两类：基于规则的方法、基于统计与学习的方法。本文将缩写识别问题转化为机器学习中的有监督分类问题，有效规避了基于规则方法中繁琐的规则制定和基于共现的统计方法中对低频词识别效果欠佳的问题。

计算机科学领域的缩写应用广泛，识别该领域学术文本中的缩写词及其对应的缩写定义有助于更好地理解并利用领域术语。本文提出的 MELearn-AI 识别算法借鉴了序列标注的思想，通过人工标注数据的方式构建训练集，在实例调查的基础上设定出 17 项分类特征，最终借助最大熵模型训练出分类器。通过对比实验，可以看出 MELearn-AI 算法相对于前人算法抽取效果更佳，尤其在查准率上得到了显著提高，但是由于学习语料不丰富的原因使其召回率提升稍有受限。

针对现有的成果和结论，笔者下一步的工作是探究一词多义带来的缩写消歧问题，并在此基础上构建计算机领域的缩写词词典，以及探索本文所提出方法在其他领域的适用性。

参考文献

[1] SOHN S, COMEAU D C, KIM W, et al. Abbreviation definition identification based on automatic precision estimates [J]. BMC bioinformatics, 2008, 9(1): 402.

[2] OKAZAKI N, ANANIADOU S and TSUJII J. A discriminative alignment model for abbreviation recognition [C] // Proceedings of the 22nd International Conference on Computational Linguistics (Coling 2008), August, 2008, Manchester. Stroudsburg, PA, USA: Association for Computational Linguistics, 2008: 657-664.

[3] WREN J D, GARNER H R. Heuristics for identification of acronym-definition patterns within text: towards an automated construction of comprehensive acronym-definition dictionaries [J]. Methods of information in medicine, 2002, 41(5): 426-434.

[4] AO H, TAKAGI T. ALICE: an algorithm to extract abbreviations from MEDLINE [J]. Journal of the American Medical Informatics Association, 2005, 12(5): 576-586.

[5] PARK Y, BYRD R J. Hybrid text mining for finding abbreviations and their definitions [C] // Proceedings of the 2001 conference on empirical methods in natural language processing, 2001: 126-133.

[6] YU H, HRIPCSAK G, FRIEDMAN C. Mapping abbreviations to full forms in biomedical articles [J].

Journal of the American Medical Informatics Association, 2002, 9(3): 262-272.

[7] XU Y, WANG Z, LEI Y, et al. MBA: a literature mining system for extracting biomedical abbreviations [J]. BMC bioinformatics, 2009, 10(1): 14.

[8] TAGHVA K, GILBRETH J. Recognizing acronyms and their definitions [J]. International Journal on Document Analysis and Recognition, 1999, 1(4): 191-198.

[9] ADAR E. SaRAD: A simple and robust abbreviation dictionary [J]. Bioinformatics, 2004, 20(4): 527-533.

[10] 刘有发 . 论现代英语缩写词的构词法 [J]. 江西社会科学 , 2002(2): 196-197.

[11] ZHOU W, TORVIK V I, SMALHEISER N R. ADAM: another database of abbreviations in MEDLINE [J]. Bioinformatics, 2006, 22(22): 2813-2818.

[12] CHANG J T, SCH TZE H, ALTMAN R B. Creating an online dictionary of abbreviations from MEDLINE [J]. Journal of the American Medical Informatics Association, 2002, 9(6): 612-620.

[13] PUSTEJOVSKY J, CASTANO J, COCHRAN B, et al. Automatic extraction of acronym-meaning pairs from MEDLINE databases [J]. Studies in health technology and informatics, 2001 (1): 371-375.

[14] 刘挺 , 车万翔 and 李生 . 基于最大熵分类器的语义角色标注 [J]. 软件学报 , 2007, 18(3): 565-573.

[15] HISAMITSU T, NIWA Y. Extracting useful terms from parenthetical expressions by combining simple rules and statistical measures [G]. Recent Advances in Computational Terminology. Amsterdam: John Benjamins Publishing Co, 2001: 209-224.

[16] OKAZAKI N, ANANIADOU S. A term recognition approach to acronym recognition [C] // Proceedings of the COLING/ACL on Main conference poster sessions. Stroudsburg, PA, USA: Association for Computational Linguistics, 2006: 643-650.

[17] NADEAU D, TURNEY P. A supervised learning approach to acronym identification [G]. Advances in Artificial Intelligence. Springer Berlin Heidelberg, 2005: 319-329.

[18] DANNELLS. Automatic acronym recognition [C] // Proceedings of the Eleventh Conference of the European Chapter of the Association for Computational Linguistics: Posters & Demonstrations, 2006. Trento: Association for Computational Linguistics, 2006: 167-170.

[19] LIU H, FRIEDMAN C. Mining terminological knowledge in large biomedical corpora [C] // Pacific symposium on biocomputing, January 3-7, 2003, Kauai, Hawaii. World Scientific, 2002: 415-426.

[20] S NCHEZ, ISERN D. Automatic extraction of acronym definitions from the Web [J]. Applied Intelligence, 2011, 34(2): 311-327.

[21] 杨巍纳 . 数学论文中常用的英文缩写词 [J]. 编辑学报 , 2006, 18(2): 121-122.

[22] SHANNON C. A Mathematical Theory of Communication [J]. Bell System Technology Journal, 1984, 27: 379-423.

[23] BERGER A L, PIETRA V J D, PIETRA S A D. A maximum entropy approach to natural language processing [J]. Computational linguistics, 1996, 22(1): 39-71.

[24] 刘岩，韩瑶 . 英语缩略语常见构词方法 [J]. 语言学研究 , 2013 (5): 253-254.

[25] 杨巍纳 . 科技英文缩写词构词方法之我见 [J]. 科技与出版 , 2006 (2): 46-48.