

基于规则的大规模试卷文本语块识别方法的研究^{*}

郭凯红^{1, 2}, 李文立¹

(1 大连理工大学 管理学院, 辽宁 大连 116024 2 辽宁大学 信息学院, 沈阳 110036)

摘 要: 提出了一种基于规则的试卷文本语块识别方法, 有效解决了试题库中大规模试题数据的初始化问题。通过定义文本语块识别规则, 构建自动机识别模型, 在理论上描述了试卷文本的识别过程。实验表明, 该模型具有良好的性能, 在此基础上, 实现了一个原型系统, 通过具体的应用实例验证了该方法的可行性和有效性。

关键词: 规则; 语块; 试卷文本; 识别模型

中图分类号: TP391.43 文献标志码: A 文章编号: 1001-3695(2009)04-1391-03

Study of massive paper texts chunking based on rules

GUO Kaihong^{1, 2}, LI Wenli¹

(1. School of Management, Dalian University of Technology, Dalian Liaoning 116024, China; 2. College of Information, Liaoning University, Shenyang 110036, China)

Abstract To solve the initiating of massive examination questions in database efficiently, proposed a paper texts chunking method based on rules. Defining recognition rules of paper texts and constructing automata recognition model described the recognition processing of paper texts theoretically. Experiment results show that this model has better performance. By these works, implemented a prototype system, and verified its feasibility and effectiveness by a practical application.

Key words rules; chunk; examination paper texts; recognition model

0 引言

浅层句法分析 (shallow parsing), 也称做部分句法分析 (partial parsing) 或语块分析 (chunk parsing)^[1], 是近年来自然语言处理领域出现的一个新的语言处理策略。它是与完全句法分析相对的, 不要求得到完全的句法分析树, 只要求识别其中的某些结构相对简单的成分, 如非递归的名词短语、动词短语等。这些识别出来的结构通常被称做语块 (chunk)。概括起来, 句法分析的方法基本上可以分成两类^[2]: 基于统计的方法和基于规则的方法。统计方法主要来自概率统计和信息论, 而规则方法则是根据人工书写的或 (半) 自动获取的语法规则标注出短语的边界和短语的类型。

浅层句法分析的主要任务是语块的识别和分析^[3], 这就使句法分析的任务在某种程度上得到简化, 同时也有利于句法分析技术在大规模真实文本处理系统中迅速得到应用, 目前已取得大量成果^[2, 4~11]。文献 [12] 从组成形式和上下文语境两个方面来识别汉语文本中的特殊符号串; 文献 [13] 提出了基于规则的数据收集策略, 将数据收集过程分成共性和特性两种情况, 将提炼出的若干规则形成不同的规则集以便于开发和定制。这些研究成果主要集中在“语块库建立”的问题上。本文面向具体应用, 研究由大规模试卷文本实现试题库高效初始化的方法, 即通过语块识别程序, 根据预先定义的识别规则, 应用规则方法对集中的大规模试卷文本进行识别, 逐级分析有限状态层叠^[14], 获取试卷文本结构及各试题的属性, 将各属性值实

时写入数据库, 自动完成试题库的初始化工作。本文方法改变了传统的人工录入试题库的操作模式, 有效解决了大规模试题库的初始化问题。

1 文本结构分析

教育考试的普通试卷 (笔试或上机考试) 主要有选择、填空、改错、问答等有限几种类型题, 每种类型题又由若干道具体的试题组成。任何一道试题总伴有题目、答案、分值等几种相关数据项, 这些实际上就是每道试题的具体信息; 将若干条这样的试题按类型分组, 就构成一组类型题; 可以将多种类型题组合在一起, 从而构成一套试卷。由此得出, 每道试题及其相关属性 (答案、分值等) 是构成试卷的基本元素, 试题库实际存储的是这些属性值, 而不是整套试卷; 每道试题通过其他相关属性标志它所属的试题类型及试卷套数。

根据以上分析, 现提取出试题必要的数据项, 即试卷序号、试题类型序号及类型说明、试题序号及题目、试题答案、试题分值。这实际上给出了试题的数据结构, 识别系统将按这种格式实现对试题数据的分析操作。

2 识别规则

根据前述的试题结构, 考虑到试卷文本的可读写性, 定义标记符号为“@”“[”“()”“#”“%”。其中: “@”是试卷开始符, 符号后必须紧跟数字, 代表试卷的序号; “[”是试题类型开始符, 符号“[”与“]”之间必须是数字, 代表试题类型的序

收稿日期: 2008-06-21; 修回日期: 2008-08-30 基金项目: 国家自然科学基金资助项目 (70572099); 辽宁省自然科学基金资助项目 (1050349)

作者简介: 郭凯红 (1973-), 男, 河南镇平人, 博士研究生, 主要研究方向为信息管理、系统工程 (guokh@126.com); 李文立 (1969-), 男, 河南平顶山人, 副教授, 博士, 主要研究方向为电子商务、信息管理、系统工程。

号;“()”是题目开始符,符号“(”与“)”之间必须是数字,代表试题的序号;“#”是试题答案开始符,符号后紧跟的内容是当前试题的答案;“%”是试题分值开始符,符号后必须紧跟数字,代表当前试题的分值。总体要求是,一套试卷必须有试卷序号、试题类型序号及类型说明、试题序号及题目内容、试题答案内容、试题分值等属性值;暂不考虑试卷文本中空格和回车换行符的识别问题,其处理方法将在实现中灵活处理。

上述标记符号被称做保留符号,如同高级语言中的保留字一样,由语块识别程序当做标记控制符号使用。如果文本中的试题内容含有上述保留符号,则将该符号连续双写,如试题中含有内容“# include <stdio.h>”,在试卷文本中应写成“## include <stdio.h>”,语块识别程序在分析这段文本时,自动地将符号串“##”解析成输出符号“#”,而不是标记控制符号。满足本识别规则的试卷文本如下所示:

```
@ 1
[ 1]选择题 1
(1)下面哪种操作系统是非图形化的?
A. Linux      B. DOS
C. Windows   D. Macintosh
#B
% 2
(2)下面哪种软件用于专业图像编辑?
A. Word       B. WPS
C. Photoshop  D. Borland C++
#C
% 2
.....
[ 4]简答题
(1)什么是计算机病毒?
#关于计算机病毒目前没有一个公认的定义,我国公安部
计算机安全检察司对病毒的定义是:计算机病毒是指编制或者
在计算机程序中插入的破坏计算机功能或者毁坏数据,影响计
算机使用,并能自我复制的一组计算机指令或者程序代码。
% 5
.....
```

3 自动机识别模型

根据前述定义的识别规则,即通过增加标记定义文本语块的边界,现建立文本语块的自动机识别模型,描述试卷文本的识别过程。首先构造机器 M 的输入字母表和输出字母表。由于试卷文本是建立在 ASCII 码上的字符串,定义输入字母表为全体可视 ASCII 码集,用 Σ 表示,即 $S = \Sigma$ 。定义输出字母表时,要求一个输入串(试卷文本)的响应在逻辑上具有可识别的特征,也就是说,建立在输出字母表上的字符串能够抽象地表示清楚每套试卷及每道试题的逻辑结构。这里用 0 表示试卷开始,当连续输出两个以上 0 时,表示输入串(试卷文本)被 M 拒绝;1 表示文本识别成功,输入串(试卷文本)被 M 接受; * 抽象地表示试题内容; [,], (,), #, % 表示识别规则中的标记符号,即标记符号依照原样输出。所以定义输出字母表为 $R = \{0, 1, *, [,], (,), \#, \% \}$ 。

下面考察 M 至少需要多少个状态。

1)初态 A M 从状态 A 出发,这是试卷文本识别过程的开始,在状态 A 下输入试卷文本的第一个符号。

2)陷阱状态 B 陷阱状态用做在其他状态下发现输入串不可能是该自动机所识别的内容时所进入的状态,在此状态下,自动机读完输入串中剩余的字符。如果输入串(试卷文本)的第一个符号是非 @, 此文本必被 M 所拒绝,此时 M 进入状态 B , 输出 Q 。一旦 M 进入状态 B , 无论输入什么符号,输出都为 Q 并且不能再转向其他状态。

3)强制接受试卷序号状态 C M 进入状态 C , 表示强制要求试卷必须有数字序号。如果输入串(试卷文本)的第一个符号是 @, M 进入状态 C , 输出 Q 表示试卷开始;继续输入的符号如果是数字 1~9 则 M 进入状态 C' , 输出 *; 否则 M 进入状态 B , 输出 Q 。

4)接受试卷序号状态 C' M 进入状态 C' , 表示输入的试卷序号暂时可以被接受。继续输入的符号如果是数字 0~9 则 M 仍处于状态 C' , 输出 *; 继续输入的符号如果是 [, 则 M 进入状态 D , 输出 [, 否则 M 进入状态 B , 输出 Q 。

5)强制接受试题类型序号状态 D M 进入状态 D , 表示强制要求试题类型必须有数字序号。继续输入的符号如果是数字 1~9 则 M 进入状态 D' , 输出 *; 否则 M 进入状态 B , 输出 Q 。

6)接受试题类型序号状态 D' M 进入状态 D' , 表示输入的试题类型序号暂时可以被接受。继续输入的符号如果是数字 0~9 则 M 仍处于状态 D' , 输出 *; 继续输入的符号如果是], 则 M 进入状态 D'' , 输出]; 否则 M 进入状态 B , 输出 Q 。

7)强制接受试题类型说明状态 D'' M 进入状态 D'' , 表示强制要求试题类型必须有文本说明。继续输入的符号如果是非 (, 则 M 进入状态 $D \odot$ 输出 *; 否则 M 进入状态 B , 输出 Q 。

8)接受试题类型说明状态 $D \odot$ M 进入状态 $D \odot$ 表示输入的试题类型文本说明暂时可以被接受。继续输入的符号如果是非 (, 则 M 仍处于状态 $D \odot$ 输出 *; 继续输入的符号如果是 (, 则 M 进入状态 E , 输出 (。

9)强制接受试题序号状态 E M 进入状态 E , 表示强制要求试题必须有数字序号。继续输入的符号如果是数字 1~9 则 M 进入状态 E' , 输出 *; 否则 M 进入状态 B , 输出 Q 。

10)接受试题序号状态 E' M 进入状态 E' , 表示输入的试题序号暂时可以被接受。继续输入的符号如果是数字 0~9 则 M 仍处于状态 E' , 输出 *; 继续输入的符号如果是), 则 M 进入状态 E'' , 输出); 否则 M 进入状态 B , 输出 Q 。

11)强制接受题目状态 E'' M 进入状态 E'' , 表示强制要求题目必须有文本内容。继续输入的符号如果是非 #, 则 M 进入状态 $E \odot$ 输出 *; 否则 M 进入状态 B , 输出 Q 。

12)接受题目状态 $E \odot$ M 进入状态 $E \odot$ 表示输入的题目内容暂时可以被接受。继续输入的符号如果是非 #, 则 M 仍处于状态 $E \odot$ 输出 *; 继续输入的符号如果是 #, 则 M 进入状态 F , 输出 #。

13)强制接受试题答案状态 F M 进入状态 F , 表示强制要求试题答案必须有文本内容。继续输入的符号如果是非 %, 则 M 进入状态 F' , 输出 *; 否则 M 进入状态 B , 输出 Q 。

14)接受试题答案状态 F' M 进入状态 F' , 表示输入的试题答案暂时可以被接受。继续输入的符号如果是非 %, 则 M 仍处于状态 F' , 输出 *; 继续输入的符号如果是 %, 则 M 进入

5 系统实现架构分析

数据库和模型库的分析是从系统实现的逻辑结构来分析的。系统的实现架构指的是以何种物理组成方式实现系统, 各数据库位于何处、模型库位于何处、以何种网络形式进行连接等。对于具有一定地域覆盖度的系统而言, 通常流行的架构方式有浏览器/服务器(B/S)、客户机/服务器(C/S)等。如果 GLD_DSS 采用的是 B/S 这类瘦客户端的逻辑结构, 那么所有的信息必须位于调度中心服务器上, 这要求连接各工作站的网络的带宽足够大。其优点是系统易于维护和管理, 各用户只通过浏览器进入系统, 用户端无须安装任何软件。对于小型粮食企业, 由于其通常只有一个粮食仓储中心, 建成高速局域网也较为简单, 适合采用这种方式。对于大型粮食物流企业, 由于其地域覆盖较广, 业务量也较大, 采用 C/S 结构较为合理。

6 结束语

本文从逻辑和实现角度对基于 GIS 的粮食物流配送决策支持系统 (GLD_DSS) 进行了总体分析。结合 GIS 的信息需求, 对数据库系统进行了较为细致的分析; 结合了 GLD_DSS 的主要决策问题, 对模型库的功能进行了分析。此外, 还考虑了系统的实现架构。粮食物流配送决策支持系统是一个较为复杂的系统, 本文所作的分析只是宏观性的, 在具体的实现过程中还要更进一步地进行细化。由于此系统涉及面宽, 实际情况比较复杂, 与粮食企业的经济效益有较大的联系, 需要不断进行更加深入的研究和开发。

参考文献:

[1] 孙吉贵, 白洪涛, 于海鸿, 等. 粮食调拨决策支持系统的设计与实

(上接第 1393 页) 统作为湖北省教育考试院组织实施的“高等教育自学上机考试系统 (for Windows)”之核心子系统, 负责后台数据库大规模试题数据的初始化工作。该软件在上机考试系统中成功应用, 并在试卷文本语块识别过程中表现出良好的性能。

6 结束语

本文给出了一种基于规则的大规模试卷文本语块识别方法及识别模型, 并以该模型为基础实现了一个原型系统, 较好地解决了数据库中大规模试题数据的高效初始化问题。实验表明, 该识别模型具有良好的性能, 对于试卷在未来可能发生的各种变化, 如题型改变或题量增减, 模型依然有较强的适应性。限于篇幅, 本文未给出模型实现的源码。由于系统使用之前必须预先了解试题库的数据结构及类型, 对于未知结构的关系表, 目前尚未考虑涉及动态获取并绑定表属性等问题, 在一定程度上限制了本系统的推广及应用范围。这将是下一步研究工作的重点。

参考文献:

[1] ABNEY S P. Parsing by chunks[M]//BERWICK R, ABNEY S, TENNY C, et al. Principle-based parsing. Dordrecht: Kluwer Academic Publishers, 1991: 257-278.
[2] SANG E F T K, BUCHHOLZ S. Introduction to the CoNLL-2000 shared task: chunking[C]//Proc of the 2nd Workshop on Learning Language in Logic. Morristown: Association for Computational Linguistics, 2000: 127-132.
[3] 孙宏林, 俞士汶. 浅层句法分析方法概述[J]. 当代语言学, 2000, 2

现[J]. 吉林大学学报: 信息科学版, 2005, 23(1): 81-85.
[2] 鹿应荣. 粮食物流系统优化研究[D]. 长春: 吉林大学, 2007.
[3] HARPER P R, SHAHANIA K. A decision support system for the care of HIV and AIDS patients in India[J]. European Journal of Operational Research, 2003, 147(1): 187-197.
[4] BERGEY P K, RAGSDALE T C, HOSKOTEM. A decision support system for the electrical power districting problem[J]. Decision Support Systems, 2003, 36(1): 1-17.
[5] HAASTRUP P, MANEZZO V. A decision support system for urban waste management[J]. European Journal of Operational Research, 1998, 109(2): 330-341.
[6] PAL K, PALMER O. A decision-support system for business acquisitions[J]. Decision Support Systems, 2000, 27(4): 411-429.
[7] WATTAU C I, AKOKA J. Logistics information system auditing using expert system technology[J]. Expert Systems with Applications, 1996, 11(4): 463-473.
[8] ALDIN N, STAHERE F. Electronic commerce: marketing channels and logistics platforms: a wholesaler perspective[J]. European Journal of Operational Research, 2003, 144(2): 270-279.
[9] 唐孝飞, 孙壮志, 胡思继. 物流配送决策支持系统的分析[J]. 北方交通大学学报, 2002, 26(5): 92-97.
[10] 陈述彭, 鲁学军, 周成虎. 地理信息系统导论[M]. 北京: 科学出版社, 2000.
[11] ZHANG Qiu-wen, ZHEN Tong, ZHU Yu-hua, et al. A hybrid intelligent algorithm for the vehicle routing with time windows[C]//Proc of International Conference on Intelligent Computing. Berlin: Springer-Verlag, 2008: 47-54.
[12] 甄彤, 张秋闻, 马志. 基于改进蚁群算法的粮食物流调度研究[J]. 河南工业大学学报: 自然科学版, 2008, 29(3): 62-65.
[13] MOYNIHAN G P, SRAJSTERING J U P, et al. Decision support system for strategic logistics planning[J]. Computer in Industry, 1995, 26(1): 75-84.

(2): 74-83
[4] ARGAMON S, DAGAN I, KRYMOLOWSKIY. A memory-based approach to learning shallow natural language patterns[C]//Proc of the 36th Annual Meeting of the Association for Computational Linguistics. Morristown: Association for Computational Linguistics, 1998: 67-73.
[5] ZHANG Tong, DAMERAU F, JOHNSON D. Text chunking based on a generalization of window[J]. Journal of Machine Learning Research, 2002, 2: 615-637.
[6] 周强, 孙茂松. 汉语句子的组成分析体系[J]. 计算机学报, 1999, 22(11): 1158-1165.
[7] 梁颖红, 赵铁军. 基于关联度评价的中心词扩展的英语文本语块识别[J]. 计算机研究与发展, 2006, 43(1): 153-158.
[8] 梁颖红, 赵铁军, 于浩. 基于改进 K-均值聚类的汉语语块识别[J]. 哈尔滨工业大学学报, 2007, 39(7): 1106-1109.
[9] 魏伟, 杜金华, 徐拔. 基于分层语块分析的统计翻译研究[J]. 中文信息学报, 2007, 21(5): 87-90.
[10] 秦玉平, 王秀坤, 艾青, 等. 多主题文本分类的实现算法[J]. 计算机工程, 2008, 34(2): 190-192.
[11] 潘大志, 成琥, 黄青松. 基于规则、串频统计和上下文关系的现代汉语分词系统的实现[J]. 内蒙古师范大学学报: 自然科学版, 2008, 37(1): 71-74.
[12] 李宏乔, 樊孝忠. 汉语文本中特殊符号串的自动识别技术[J]. 计算机工程, 2004, 30(12): 114-115.
[13] 陈永府, 杨小献, 黄正东. 基于规则的数据收集研究[J]. 计算机工程与设计, 2007, 28(1): 158-161.
[14] ABNEY S. Partial parsing via finite-state cascades[J]. Natural Language Engineering, 1996, 2(4): 337-344.