汉语文本的篇章结构及其标引算法的研究

单 永 明

山西大学计算机科学系，太原 0 3 0 0 0 6

摘要： 本文从形式化的角度讨论了汉语文本的篇章结构，给出了文本的标题、子标题、段落及其层次结构的一种划分与标记方法，提出了规范的与准规范文本的概念，并以此为基础讨论了篇章结构的标引算法。本文阐明的观点及方法和结果对汉语文本的全文文本标引及结构化分析具有直接的现实意义。

关键词： 中文信息处理，文本自动分析，篇章结构，标引树，自动标引算法。

Study of Chinese Writings Structure and Its Indexing Algorithm

Shan Yongming

Department of Computer Science,Shanxi University,Taiyuang 030006

ABSTRACT: In the paper,we discuss chinese writings structure from the point of view of formalization. Formal descriptions for the heading, subheading and paragraph as well as their structural relations in a text are present and a systematic tagging method for chinese writings structure is proposed. And then, we introduce the conceptions of the normal text and quasi-normal text. On the basis of these, automatic indexing algorithm for chinese writings structure is discussed. The standpionts, as well as the methods and results presented in the paper are of direct and practical significance for fulltext indexing and structural analyses and processes of chinese text.

Keywords : Chinese information processing, automatic text analyses, writings structure, indexing-tree, automatic indexing algorithm.

在汉语文本的自动标引中，各标题、自然段、图、表等在文本中所处的位置，对文本自动分析、自动分类、自动文摘等的有效进行是十分重要的[2-4]，为此本文在文[1]的基础上就汉语文本的篇章结构标引问题作了进一步的讨论，给出了文本的标题、子标题、段落及其层次结构的一种划分与标记方法，并提出了规范文本与准规范文本的概念，最后给出了规范与准规范两类文本的篇章结构自动标引算法。

1．文本及其篇章结构的形式描述

我们将汉语文本（以下简称文本）中的标题（包括文本的题目）、自然段等看做单位子串。若以 w 表示文本中的单位子串，符号“｜”表示文本中的“另起一行”运算，则我们可给出文本的形式定义如下：

定义 1　文本

一个文本是由若干单位子串经｜运算而得到的线性序列。由 n 个（n≥1）单位子串经｜运算而得到的文本 text 可表示为 ：text＝w_0｜w_1｜w_2．．．｜w_{n-1}。其中 w_i（0≤i≤n-1）称为 text 的第 i 个单位子串。

文本 text 的长度定义为 text 中单位子串的个数，记作｜text｜。若 w 是 text 的一个单位子串，则记作 w∈text。

虽然单位子串的类型不止于标题和自然段，但在下面的讨论中，为了简化和更有利于问题实质的探讨，我们将只就仅由标题和自然段作为单位子串组成的文本进行讨论，并且我们将看到，所得到的结论仍不失一般性。

在文本中，文本的题目表示文本的开始；一个标题表示一个章节的开始，而一个章节又可由若干子章节组成，每个子章节又有自己的小标题，我们称这些小标题为直接包含它的章节的标题的子标题，而称一个自然段为直接包含它的章节的标题的子段。若 w_{ki}（i＝1，2，...）同是标题 w_k 的子标题，则按其在文本中出现的先后顺序，分别称其为 w_k 的第一子标题、第二子标题...等；对于子段，亦有相应的名称：w_k 的第一子段、第二子段...等。

基于上面的论述，我们引入以下概念。

定义 2　标题的级，子段的级

文本 text 中标题或子段的级递归定义如下：

（1）文本的题目 w_0 的级定义为 d（w_0）＝0

（2）若 w_{ki} 是标题 w_k 的子标题或子段，则 w_{ki} 的级定义为 d（w_{ki}）＝d（w_k）+1。

定义 3　偏序$<_p$

在文本 text 上定义偏序$<_p$如下：$w_{ki}<_p w_{kj}$ 当且仅当同时满足下述条件：

（1）w_{ki}、w_{kj} 是同一标题 w_k 的子标题或 / 和子段，且 $k_i<k_j$；

（2）若 w_{kl} 也是同一标题 w_k 的子标题或子段（$k_l\neq k_i$，$k_l\neq k_j$），则要么有 $k_l<k_i$，要么有 $k_j<k_l$。

标题或子段的级描述了文本 text 中各标题或子段的层次关系。偏序$<_p$描述了文本 text 中任一标题的子标题和子段在 text 中出现的先后顺序。设 w_{ki}（1≤i≤m）是 w_k 的第 i 个子标题或子段，则 w_k 的全部子标题和子段的偏序是 $w_{k1}<_p w_{k2}<_p w_{k3}...<_p w_{km}$。

下面引入文本的 index 标记及标引树的概念。

定义 4　文本的 index 标记

文本 text 的 index 标记是按下述方法得到的标题或子段的 index 标记的集合：

（1）文本 text 的题目 w_0 的 index 标记为 h-index（w_0）＝h-0；

（2）若 w_k∈text，w_k 是标题，其 index 标记为 h-index（w_k），且 w_k 有 m 个子段和子标题 w_{ki}（1≤i≤m），则 w_{ki} 的 index 标记是：若 w_{ki} 是 w_k 的第 i 个子段，则 w_{ki} 的 index 标记是 p-index（w_{ki}）＝p-index（w_k）·i（称为 p 型 index 标记，p 是自然段的 index 标记的前缀）；若 w_{ki} 是 w_k 的第 i 个子标题，则 w_{ki} 的 index 标记是 h-index（w_{ki}）＝h-index（w_k）·i（称为 h 型 index 标记，h 是标题的 index 标记的前缀）。

定义 5　文本 text 的标引树 T_I（text）

文本 text 的标引树 T_I（text）（当 text 已明确时，简记为 T_I）是一棵满足下述条件的具有 index 标记的树：

（1）标引树 T_i 的根结点是文本 text 的题目 w_0，其标记为 w_0 的 index 标记；

（2）设标记为 index（w_k）的结点 w_k 是文本的一个标题，若该标题有 m 个子标题和子段 w_{k1}、w_{k2}、... w_{km}，则该结点有 m 个子结点，这些子结点与子标题和子段的对应关系是：若 m 个子标题和子段的偏序是 $w_{k1}<_p w_{k2}<_p w_{k3}$ . . . $<_p w_{km}$，则 m 个子结点从左至右依次是 w_{k1}、w_{k2}、. . . w_{km}，且 w_{ki}（$1\leqslant i\leqslant m$）结点的标记是 w_{ki} 的 index 标记。

图 1 是一棵标引树的例子。在标引树中，父结点和子结点之间具有层次关系（从上至下），兄弟结点之间具有顺序关系（从左至右），因此，文本的标引树准确描述了文本的线性序列所蕴含的结构特征。

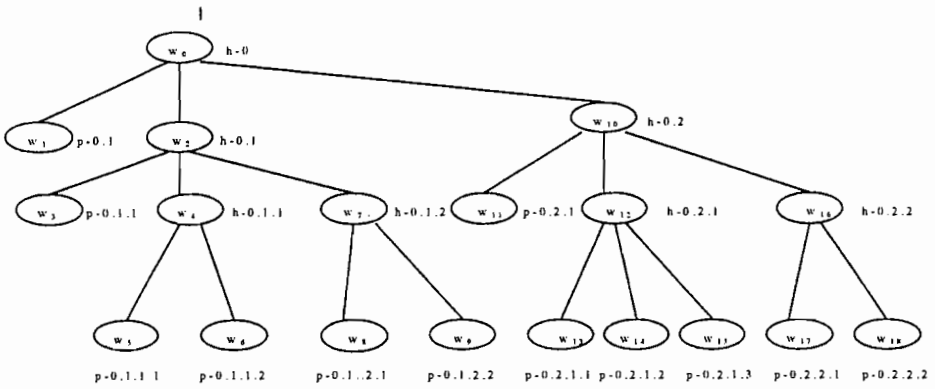

图 1． 一棵标引树.

2．规范的和准规范的文本

为了得到对文本的篇章结构进行标引的有效算法，下面讨论规范文本、准规范文本的概念。

定义 6 标题符号集

设 $w_k\in$text，且是一个标题，则 w_k 可看作 t、u、e 经字连接运算（以·表示）而得的串，即 w_k=t·u·e，其中：t 是标题符号，u 是具有实际意义的汉字字符串，e∈{ε，！，?}。由文本 text 的全部标题符号组成的集合称作 text 的标题符号集，记作 Title={t | t 是 text 中的标题符号}。

对一个文本所使用的标题符号集 Title，我们可根据其中标题符号的型式和使用功能将它划分为若干个子集 $title_i$（i=1,2,3,...），使得 Title 中的每个标题符号属于且仅属于其中的一个子集，那么这些子集的全体构成的集合称作 Title 的一个型划分，其中的每一个子集称作 Title 的一个型。在同一个型中的标题符号称为同型标题符号。

定义 7 标题的型，同型标题，非同型标题

设 w_k=t· u·e 是 text 的一个标题，若存在型 $title_i\subseteq$Title，使得 t∈$title_i$，则称 $title_i$ 为标题 w_k 的型，记作 type（w_k）=[$title_i$]。设有 w_i、w_j 是 text 中的两个标题，若 type（w_i）=type（w_j），则称 w_i 与 w_j 为同型标题，否则称 w_i 与 w_j 为非同型标题。

根据定义 2 及定义 7，我们引入下述概念。

定义 8　规范的、准规范的和非规范的标引树

（1）在一棵标引树中，若满足下述条件：①对任一非叶结点 w_k，设 w_k 有 m 个子结点 $w_{k1} <_p w_{k2} \ldots <_p w_{km}$，则有：若 w_{kl}（$1 \leqslant l \leqslant m$）具有 p 型 index 标记，则 w_{ki}（$1 \leqslant i < l$）均具有 p 型 index 标记；若 w_{kl}（$1 \leqslant l \leqslant m$）具有 h 型 index 标记，则 w_{kj}（$l < j \leqslant m$）均具有 h 型 index 标记；②对任意两个具有 h 型 index 标记的结点 w_i、w_j 有：type（w_i）＝type（w_j）当且仅当 d（w_i）＝d（w_j），则称该标引树是规范的。

（2）在一棵标引树中，若其任一棵子树都满足：①对任一非叶结点 w_k，设 w_k 有 m 个子结点 $w_{k1} <_p w_{k2} \ldots <_p w_{km}$，则有：若 w_{kl}（$1 \leqslant l \leqslant m$）具有 p 型 index 标记，则 w_{ki}（$1 \leqslant i < l$）均具有 p 型 index 标记；若 w_{kl}（$1 \leqslant l \leqslant m$）具有 h 型 index 标记，则 w_{kj}（$l < j \leqslant m$）均具有 h 型 index 标记；② 对于该子树第一层中的任意两个具有 h 型 index 标记的结点 w_i、w_j，均有 type（w_i）＝type（w_j）；③在该子树及其祖先结点中，对于任意两个具有 h 型 index 标记的结点 w_i、w_j，若 d（w_i）≠d（w_j），则 type（w_i）≠type（w_j），则称该标引树是准规范的。

（3）若一棵标引树不是规范的和准规范的，　则称之为非规范的。

若一个文本的标引树是规范的（准规范的，非规范的），　我们亦称该文本是规范的（准规范的，非规范的）。对于规范的和准规范的标引树，我们有下述结论：

定理 1　若一棵标引树是规范的或准规范的，　则从根结点到任一叶结点所形成的路中，对任意两个具有 h 型 index 标记的结点 w_i、w_j，都有 type（w_i）≠type（w_j）。

基于定理 1，对于规范的和准规范的文本，　我们可以有效地识别和标识文本中标题和自然段的级。通常情况下，我们所遇到的文本，大多数都是规范的这一类文本。

3．标引算法

汉语文本篇章结构的标引，就是要通过对文本这一线性序列的扫描，　确定各单位子串的类型及在篇章中的位置，并给出相应的标记。通过上面几节的讨论知，　一个文本的标引树唯一确定了该文本的篇章结构；若文本是规范的或准规范的，则标引树中任一结点的 index 标记唯一确定了对应单位子串的类型及在篇章中的位置。因此，在实际标引过程中，对规范或准规范的文本，只要给出每个单位子串的 index 标记，就达到了篇章结构标引的目的。

基于上述定理 1，下面我们就规范的和准规范的文本给出两个标引算法。

在下述算法中，我们将使用一个标引栈 IS 存放有关被标引结点的数据，　其结构如图 2 所示。IS 栈的每一项可表示为一个四元组

（t，Typeh，Indexh，Indexp ）

其中：t 是栈顶指针，t＝0，1，2，...；Typeh 是当前被标引过的标题 w_k 的型 type（w_k）；　Indexh 是当前被标引过的标题在当前层的 index 标记分量，　Indexh∈N∪｛0｝；Indexp 是当前被标引过的子段在当前层的 index 标记分量，　Indexp∈N；N＝｛1，2，...｝。开始时，　栈顶内

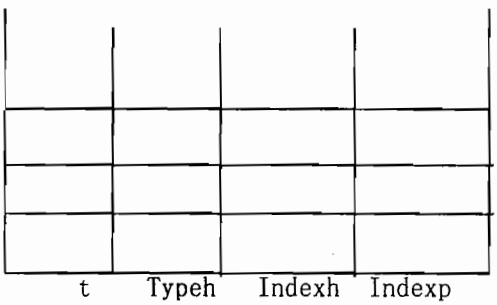

图 2.　标引栈 IS.

容为（0，∧，0，0）。若某一时刻，栈顶内容为（I，[title$_j$]，j_i，i_i），则表示当前被标引过的子标题是T_i中第 I-1 层某一标题的第 j_i个子标题，标题的型为[title$_j$]；或表示当前被动标引过的子段是 T_i中第 I-1 层某一标题的第 i_i个子段。

在算法中使用的几个函数如下：

READ（w）：从文本指针当前位置开始读一个单位子串到变量 w，指针后移到下一个单位子串前。

IDENTIFY（w）：识别函数。若 w 是标题，返回值为 h；若 w 是自然段，返回值为 p。

TYPE（w）：取标题 w 的型作为返回值。

INDEX（w, IS.Indexh[t]）：标引函数。将 h-IS.Indexh[0]· IS.Indexh[1]· IS.Indexh[2]·...IS.Indexh[t]作为标题 w 的标引值。或 INDEX（w, IS.Indexp[t]）：将 p-IS.Indexh[0]·IS.Indexh[1]·...IS.Indexh[t-1]·IS.Indexp[t]作为自然段 w 的标引值。即该标引函数给出的是 w 的 index 标记。

BACK：文本指针移到文本开始位置。

为简单清晰起见，算法未考虑文本无题目的情况。

算法 1　规范文本的标引

1.［初始化］

t:＝0; BACK

2.［标引文本题目］

（1）READ（w）

（2）如果 IDENTIFY（w）＝h

则① IS.Typeh[t]:＝TYPE（w）；IS.Indexh[t]:＝0; IS.Indexp[t+1]:＝0

② INDEX(w, IS.Indexh[t])

3.［每循环一次，标引一个单位子串］

循环　当文本未结束时，反复执行下列语句

（1）READ（w）

（2）如果 IDENTIFY（w）＝p

则 ① IS.Indexp[t+1]:＝IS.Indexp[t+1]+1

② INDEX（w, IS.Indexp[t+1]）

（3）如果 IDENTIFY（w）＝h

则 ① 如果对于每一个 i，1≤i≤t，均有 IS.Typeh[i]≠TYPE（w）

且（IS.Typeh[t+1]＝∧或 IS.Typeh[t+1]＝TYPE（w））

则（i）t:＝t+1

（ii）如果 IS.Typeh[t]=∧ 则 IS.Typeh[t]:＝TYPE（w）

② 如果存在某个 i，1≤i≤t，使得 IS.Typeh[i]=type（w）

则　当 t＞i 时，反复执行下列语句

（i）IS.Indexh[t]:=0; IS.Indexp[t+1]:=0

（ii）t:＝t-1

③ 如果非①、②情形，则为准规范文本。转 4

④ IS.Indexh[t]:＝IS.Indexh[t]+1; IS.Indexp[t+1]:=0; INDEX（w，IS.Indexh[t]）

4.［清栈］（略）

5.［算法结束］.

算法 2 准规范文本的标引

1. 同算法 1 步骤 1

2. 同算法 1 步骤 2

3.［每循环一次，标引一个单位子串］

循环 当文本未结束时，反复执行下列语句

（1）同算法 1 步骤 3（1）

（2）同算法 1 步骤 3（2）

（3）如果 IDENTIFY（w）=h

则 ① 如果对于每一个 i，1≤i≤t，均有 IS.Typeh[i] ≠TYPE（w）

则（i） t:=t+1

（ii）IS.Typeh[t]:=TYPE（w）

② 如果存在某个 i，1≤i≤t，使得 IS.Typeh[i]＝TYPE（w）

则 当 t＞i 时，反复执行下列语句

（i） IS.Indexh[t]:=0; IS.Indexp[t+1]:＝0; IS.Typeh[t]:＝∧

（ii）t:＝t-1

③ 同算法 1 步骤 3（3）④

4. 同算法 1 步骤 4

5.［算法结束］

4．结束语

以上我们就汉语文本的篇章结构及其标引算法进行了讨论，由讨论可知，以标引树做为篇章结构的数学模型是合理的和有效的，由此提出的文本的篇章结构分类具有相当的普遍适用性。其中规范文本是我们日常所见到的相当一大类文本。在上述讨论中，我们将文本看做是以标题和自然段作为单位子串组成的线性序列，这样的抽象和简化，有利于问题实质的描述和刻划，且并不影响实情况的处理；在实际应用中，我们还可扩充单位子串的类型，例如图、表等[1]，则本文所述观点及方法仍然有效，只是在标引树的结点标识及标引算法中，应扩大单位子串类型的标识范围，并增加相应的识别和标引功能。

参考文献

[1] 单永明. 一类规范文本篇章结构的自动标引. 中文信息学报，第 12 卷，第 4 期, 1998（4）, 47-52.

[2] 苏新宁. 汉语文献自动标引综析, 情报学报, 1993, 12(4), 309-318.

[3] 王建波 王开铸，自然语言篇章理解及基于理解的自动文摘研究，中文信息学报，1992, 6(2), 1-7.

[4] Salton G,Allen J.Automatic text Decomposition and Structuring. Information Processing & Management. Vol.32,No.2,pp127-138,1996.

[5] 韦雄观，等. 基于关系图的篇章分析方法，模式识别与人工智能，1997，10(2)，112-117.