自然言語処理　32－7
（1982. 8. 2）

日本語テキスト・データの蓄積処理について

星野雅英・田嶋一夫　（国文学研究資料館）

1．はじめに

日本語のテキスト・データを取り扱うシステムは少なくない。［1～3］　形態的なデータの蓄積・検索の立場から関連するシステムをみると，多くのシステムはテキスト内容や研究・利用目的に強く依存した形で蓄積されているようである。また，蓄積されたテキスト・データを利用者が直接的に自由に検索できるようなシステムは少なく，あってもその機能は小さく，蓄積されたデータに強く依存しているようである。

従来の索引誌作成システムや語彙調査システムはデータ・ベースへの指向が少なく，テキスト・データの検索システムではオンラインの事実検索や文献検索の一種としてとらえ，既存のＤＢＭＳやＩＲシステムを利用すれば良いという考え方が強いようである。”日本語テキスト・データ”用のＤＢＭＳやＩＲシステムが必要ではないかと思われる。このらの点については別に本研究会で発表する。［4］

国語・国文学研究にとって，大量のテキスト・データが蓄積され，検索できる必要がある。大量の”日本語テキスト・データ”用のＤＢＭＳやＩＲシステムがあれば良いが，現在この種のシステムがないこと，実現が容易でないこと等から，形態的な面から蓄積・検索できる範囲に限定したＩＲシステムの１つを開発中である。途中経過についてはその都度発表してきたが［5］，今回蓄積部分に大幅な改訂加えたのであらためて報告することにした。

2．基本的考え方

日本語のテキスト・データを作成する場合，一般的には，テキスト毎に「文」に分かち書き処理を施し，その１単位（すなわち「語」）毎に表記・読み・品詞等の「属性」を付加する。この時分かち書き処理は必ずしも１種類の単位になされるとは限らず，同時に２種類以上になされることもある。また特定の種類に固定されているとは限らない。同様のことが属性についても言える。また，第一の蓄積対象と考えている，日本語の古典文学作品のテキストでは文の種類（「文体」より広義に考えている）が複数あり，１つのテキストが１種類の文のみからなる時と複数の文からなる時がある。前者には小説類，後者には歌物語（和歌／物語文），和歌集（和歌／詞書等）等がある。問題は，文の種類によって分かち書きや付加属性の種類が異なる場合があり，しかも１つのテキストとしてまとめてファイルすべきであるということにある。

一方，検索時には，どの種類の分かち書きや属性が重要で，特にどの語が重要であるといった限定づけはできない。いずれもが検索の対象となり得るものである。同一の種類の分かち書きで連続した語，複数の種類の属性が同時に検索の対象となり得る。複数の分かち書きでの語の親子関係，文での語の位置関係もあわせて検索の対象となり得よう。

そこで，「語」に分かち書きの種類の大小関係での”レベル”と，同一種類での分かち書きでの１文中の順序関係での”位置”という概念を持たせることによって，「語」の順序関係，親子関係を表現するデータ・モデルの１つを提案する。１文毎に，文を「根」とし，各レベルの各語を「葉」とする順序関係を持った木構造を応用したものである。各レベルの語数は一定でなく，文毎に，レベル毎に異なることを考慮した。

このモデルに基いた，物理的な蓄積レコード・フォー

マットを設計し，処理システムを開発した。

本システムの特徴は，

（1）分かち書きの種類や付加属性の種類を固定化せず文の種類毎に異なっても良い方式としたことにより，分かち書きや付加属性の詳しさの程度に応じた，柔軟性のある蓄積が可能であること。

（2）「語」にレベルと位置の概念をもたせ，同一レベルでの順序関係，親子関係の表現と，「属性」を最小単位として照合できることから，様々な検索が可能であること。また検索用のサブルーチンや検索言語がデータ内容に依存せずにシステム化でき，その操作性が容易であると想定されること。

にある。

なお，古典文学のテキスト・データは，

・大量であること

・「どう表記され，どう読まれたか（読めるか）」が重要であること

等から，カナ漢字変換による入力方式や，分かち書きや読み・品詞付加等の自動処理機能の導入は時期尚早と思われる。また，特定の情報以外は辞書等を用いる方式も辞書を作ること自体が古典テキストにとってまだ負担が大きすぎるので，採用しない。特に，韻文では婉曲的な表現や掛詞などの技巧的な表現が多く，現段階では不可能であろう。従って，当面，

”分かち書き処理を施し，必要な属性を付加した上で，漢字で直接入力する”

方式が妥当とおもわれる（大量データであるので漢字入力会社に委託することを想定している）。

3．蓄積データ・フォーマット

3．1　属性の特徴

付加属性は，詳しくみると次の3種がある。

（タイプ1）表記，読みなどのように，複数の種類に分かち書きされる場合，大きい単位の属性は小さい単位の属性の集まりで　表現されるもの。

（タイプ2）品詞などのように，特定の種類の分かち書きの語にのみ付加されるもの。

（タイプ3）その他。活用語の終止形のように，特定の語に付加されるもの。

タイプ1の属性については分かち書き処理が複数の種類になされる場合，共用できる。

3．2　蓄積対象データの条件及び蓄積方針

前章及び前項で述べてきた，データの特徴を踏まえ，蓄積対象データの条件及び蓄積方針を次のように設定する。

（1）テキスト単位にまとめてファイルする。

（2）各テキストは複数の文からなり，1文を1レコードとしてまとめる。また，各テキストは1種類の以上の文からなり，文の種類毎に（3），（4）の処理が異なることがある。

（3）各文は1つ以上の種類に分かち書きされる。

（4）各分かち書きされた単位には1つ以上の種類の属性が付加される。

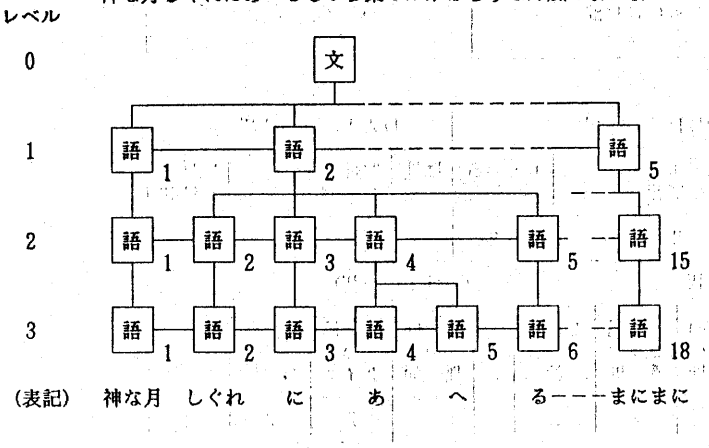
（5）複数の種類に分かち書きされる場合，必ず小さい単位は大きい単位に含まれるものとする。例外として全く分かち書きされないものも認める。

3．3　データ・モデルの提案

図1に示すような木構造を応用したデータ・モデルを提案する。分かち書きの種類で大きい単位から順次レベルを1，2，・・とし，各レベルの各語を先頭から，1，2，・・と順序づけたものである（文は0レベルとする）。各レベルでの順序関係，上下レベル間の語の親子関係が重要視されることを考慮した。レベルに一定の意味（同一レベルでは分かち書きの種類が同じであること）を持たせ，どのレベルの語も共通的に取り扱えることに特徴がある。属性は各レベルの各語に必要に応じて付加されるものとする。

文を　S，iレベルのj番目の語を　W_{ij}とすると，

図1．データ・モデル例



すべてのiに対して，

$$S = \sum_j W_{ij} \quad \cdots (1)$$

が成り立つ。語 W_{ij} は下位の語 $W_{i+1j'}$, $W_{i+1j'+1}$, , $W_{i+1j'+n}$ によって

$$W_{ij} = \sum_{j'} W_{ij'} \quad (2)$$

が成り立つ。各レベルの各語 W_{ij} は属性 A_{ijk} によって次式で表現される。

$$W_{ij} = \sum_k A_{ijk} \quad (3)$$

3．1で示したタイプ1の属性では，文の属性 S_k は任意のiに対して，

$$S_k = \sum_j A_{ijk} \quad (4)$$

で表現できる。

（1）は文と各レベルの語の関係を，（2）は上下間レベルの親子関係を，（3）は語と属性の関係を，（4）はタイプ1での文と属性の関係を表現する。

3．4データ・モデルの物理的表現

テキスト毎に，1文を1レコードとしてシーケンシャルに取り扱うことを前提とし，各レコード内を以下に示す方式で表現した。

図2にレコード・フォーマットを示す。1文を1レコードの単位とし，各レベルの各語と1対1に対応するセルを設け，このセルの集まりと各語の属性の集まりで表現する。1レコードは

（1）HEADER部

（2）WORD部

（3）DATA部

からなる。（1）はレコード情報／ID／WORD部の情報／DATA部の情報からなる。（2）はセルの集まりで，レベル1，2，・・の順で，各レベルで前から後へ順に配列されたものである。各レベルの先頭セル・末尾セル位置／セル数らの情報は（1）のWORD部の情報で示される。（3）は属性の種類毎にまとめたものである。属性の種類，属性値の長さ等の情報は（1）のDATA部の情報で示される。タイプ1の属性では，そのまま文の属性でもある。図3にセルの構造を示す。各レベルの語数，先頭・末尾位置と語の順序関係を，WORD部のレベル情報とセルの配列で表現する。語の親子関係を長男へのポインタ，親へのポインタ，子の数で表現する。親を同一とする兄弟関係を弟の有無で表現する。また，DATA部の情報，属性フラグ（有無を示す），属性iの情報で各語の属性を表現する。

本方式は，

・蓄積されたデータは修正されることが少ない（修正される場合蓄積処理から処理しなおされる）こと

・検索の即時性よりも，豊富な検索機能への要求が強いこと

・探索が同一レベルでなされれることが多く，その操作性が容易な方式を目指したこと

等による。

本方式は2進木の表現方法ほど柔軟性はないが，探索時にiレベル，j番目の語への位置づけが容易であり，蓄積されたデータ内容に依存せずシステム化でき，検索用のサブルーチンや検索言語等の開発が比較的容易だと想定される。

図2．蓄積マスタのレコード・フォーマット

HEADER部	WORD部	DATA部

(HEADER部)

項目	レコード情報	ID	WORD部の情報					DATA部の情報			
			セル総数	レベル数	レベル1の情報	・・・	レベル5の情報	属性数	属性1の情報	・・・	属性5の情報
長さ	2	18	2	2	8		8	2	8		8

(byte)

(レベルiの情報)

分かちきの種類	先頭セルの番地	末尾セルの番地	セルの数
2	2	2	2

(属性iの情報)

属性の種類	属性値のタイプ	属性値の番地	属性値の長さ
2	2	2	2

(WORD部)

項目	セル1	セル2	・・・	セルN−1	セルN
長さ	18	18		18	18

(セル総数：N)

(DATA部)

項目	属性値1	・・・・	属性値K
長さ	可変長		可変長

(属性数：K)

図3．セルの構造

項番	1	2	3	4	5	6	7	8	9	10	11	12
項目	フラグ	レベル	属性値フラグ	子の数	弟の有無	長男へのポインタ	親へのポインタ	属性の値情報1	属性の値情報2	属性の値情報3	属性の値情報4	属性の値情報5
長さ	1	3	5	10	1	12	12	20	20	20	20	20

(bit)

(属性値iの情報)

項番	8・・12	
項目	属性の値番地	属性の値長さ
長さ	12	8

4．蓄積処理

4．1　処理の概要

(1) 入力マスタの作成

初期データ・ファイルを1文の単位に分離し，必要な前処理を行って入力マスタを作成する。初期データ，入力マスタの違いは論理的なフォーマットの違いであつていずれもＫＳＬフォーマット［6］である。オンラインで校正できる。

入力マスタの文データ（ＩＤらを除いた文そのもの）では分かち書きのレベル（その区切り）と属性の種類を識別するためにそれぞれ一意なディリミタを用いる。前者を分割記号と呼び，後者を開始記号／終了記号と呼ぶ。表1に文データの文法を，図4に入力マスタの例（新勅撰和歌集）をＫＳＬフォーマットで，それぞれ示す。

(2) 蓄積マスタ作成処理

入力マスタの1レコードから，各ディリミタを取り除き，蓄積マスタの1レコードを作成する。文の種類毎にディリミタは異なって良い。ディリミタのコード（16進コードまたはＥＢＣＤＩＫコード），分かち書きの種類，属性の種類等を蓄積時のパラメタで指示することによって，データ内容に応じた蓄積処理がなされる。

図5にパラメタの例を示す。新勅撰和歌集の和歌に対する例である（文の種類毎にこれらを示すことになる）。

分かち書きの種類（STYPE，・・）を指定した順にレベルは1，2，・・となる。

表1．文データの文法

<文> ::= <語$_1$> [<分割記号$_1$><語$_1$> [<分割記号$_1$><語$_1$> [・・・]]]

<語$_i$> ::= <語$_{i+1}$> [<分割記号$_{i+1}$><語$_{i+1}$> [<分割記号$_{i+1}$><語$_{i+1}$> [・・・]]] (1)

| <語$_{i+1}$> [<分割記号$_{i+1}$><語$_{i+1}$> [<分割記号$_{i+1}$><語$_{i+1}$> [・・・]]]

<分割記号$_{i+1}$><属性$_j$> [<属性$_{j+1}$>] ・・・ [<属性$_5$>] ($j \geq 2$) (2)

| <属性$_1$> [<属性$_2$>] ・・・ [<属性$_5$>] (3)

<属性$_j$> ::= <開始記号$_j$><属性値><終了記号$_j$>

<分割記号$_i$> ::= ‖ | | | : | ・・・

<開始記号$_j$> ::= φ | (| < | [| ・・・

<終了記号$_j$> ::= φ |) | > |] | ・・・

<属性値> ::= 神な月 | かみなづき | 名詞 | ・・・ (φ：空列)

(注) (1)：下位レベルの<語$_{i+1}$>が有る時（ただし（2）を除く）。
(2)：下位レベルの<語$_{i+1}$>が有り，かつそのレベルでの<属性 >が有る時。
(3)：最下位レベルの時。 ($i, j = 1, 2, \cdots, 5$)

図4．入力マスタ例

(行番号) (漢字データ)

```
00003200 ¥A  N0000022★
00003210 山辺(やまべ)/赤人(あかひと)★
00003220 ¥W  N0000022★
00003230 山もと(やまもと)<10>/に(に)<80>‖ゆき(ゆき)<10>/は(は)<80>/ふ(ふ):り(★
00003240 り):<31>/つゝ(つつ)<80>‖しかすがに(しかすがに)<21>‖こ(こ)<11>/の(の)<★
00003250 80>/かはやなぎ(かはやなぎ)<10>‖も(も):え(え):<35>/に(に)<70>/ける(ける★
00003260 )<70>/かも(かも)<80>‖★
00003270 ¥K  N0000023★
00003280 柳(やなぎ)<10>/を(を)<80>/よ(よ):み(み):<31>/侍(はべり)<70>/ける(け★
00003290 る)<70>/★
00003300 ¥A  N0000023★
00003310 伊勢(いせ)★
00003320 ¥W  N0000023★
00003330 あをやぎ(あをやぎ)<10>/の(の)<80>‖えだ(えだ)<10>/に(に)<80>/かゝ(かか)★
00003340 :れ(れ):<31>/る(る)《り》<70>‖はるさめ(はるさめ)<10>/は(は)<80>‖いと(★
00003350 いと)<10>/もて(もて)<80>/ぬ(ぬ):け(け):<35>/る(る)《り》<70>‖たま(た★
00003360 ま)<10>/か(か)<80>/と(と)<80>/ぞ(ぞ)<80>/見(み):る(る):<32>‖★
```

(注) 行番号 3200,3220,3270,3300,3320：ＩＤ情報
3210,3300：文データ（著者）
3230-3260,3330-3360：文データ（和歌）
3280-3290：文データ（詞書）
★マークは物理的なレコードの終りを示す。

‖：句の区切り
／：語の区切り
：：語幹・語尾の区切り
( )：読み
< >：品詞

図5．蓄積パラメタ例

```
SAKU='004';
BTYPE='W ';
 STYPE='01',SEP='A1C2';
 STYPE='04',SEP='/';
 STYPE='06',SEP=':';
 ATYPE='10',SEP1=' ',SEP2=' ';
 ATYPE='20',SEP1='(',SEP2=')';
 ATYPE='30',SEP1='A1D2',SEP2='A1D3',
            DTYPE='2',DLEVEL='2';
 ATYPE='32',SEP1='A1D4',SEP2='A1D5',
            DTYPE='3';
 ATYPE='11',SEP1='A1DA',SEP2='A1DB',
            DTYPE='3';
```

(注) SAKU：作品ナンバー（新勅撰和歌集）
BTYPE：文の種類（和歌）
STYPE：分かち書きの種類
SEP：分割記号（コード）
ATYPE：属性の種類
SEP1：開始記号（コード）
SEP2：終了記号（コード）
DTYPE：属性値のタイプ
DLEVEL：属性値のレベル

4．2　蓄積処理例

4点の和歌集を処理した。表2に新勅撰和歌集での蓄積結果例を，表3に4点の蓄積容量と処理時間を示す。和歌集では蓄積容量が4～5メガバイトで処理時間が30～40分程度の範囲内であるという目安がついた。

表2．新勅撰和歌集での蓄積処理例

文の種類	文数	平均語数	最大語数	長さ K byte	平均長 byte	最大長K byte	分かち書きの種類	属性の種類
和歌	1382	16.5	296	1432	1035	16.8	01,04,06	10,20,30,32,11
詞書	966	8.7	90	358	371	2.7	04	10,20,30,32,11
著者	1231	2.0	5	238	193	0.3	04	10,20,11,21
序文	9	45.9	148	13	1469	4.4	04	10,20,30,32,11
分類	62	0	0	7	115	0.1		10
計	3650	9.5		2047	561			

(注) ＊語数は分かち書きが04のもの。
＊分かち書きの種類：01--句，04--語，06--語幹・語尾
＊属性の種類：10--表記，20--読み，30--品詞
11--統一表記，21--統一読み，32--連用形

表3．蓄積容量と処理時間例

作品名	文　数	容量 M byte	分かち書きの種類	属性の種類	処理時間
万葉集	7220 (4511)	4.5	01,04	10,20	34分25秒
古今和歌集	2890 (1111)	1.2	01,04	10,20	8分14秒
新古今和歌集	6104 (2013)	2.1	01,04	10,20	13分29秒
新勅撰和歌集	3650 (1382)	2.0	01,04,06	10,20,30,32,11	18分29秒

(注) ＊文数の括弧内は和歌の数である。
＊分かち書きの種類，属性の種類は「和歌」での例をしめす。
＊処理時間は国文学研究資料館・電子計算機システムでのCPU TIME。

5．　まとめ

形態的な検索を目的とした，日本語の古典テキスト・データの蓄積のための表現と処理システム及び処理例について述べた。大量データであるため容量が大きく，処理時間も多くかかるが，一応妥当な範囲で，データ内容に応じた柔軟性のある蓄積が可能なシステムを実現できた。

意味的な取り扱いまで及ばなければ，オリジナルなテキスト・データの蓄積システムとして十分寄与するものと考えられる。

ただ，分かち書きのレベルに一定の意味を持たせ，そのレベル・属性数ともに制限したことに一定の限界がある。より柔軟性のあるシステムへの検討が必要である。

また，データ作成時に自動分かち書きや品詞等の自動付加等の機能を利用していくことが今後の大きな課題である。

同時に語彙調査や公開利用への展開時には，ある程度のデータの統一や制限等が必要になり，これらへの対処も今後の課題である。

現在データの整備と検索用のサブルーチン／簡易型検索言語／検索機能を持った索引誌作成システム等を開発している。

なお，本研究の一部は昭和55～56年度文部省科研費(NO．581009)によった。

(参考文献)
1）長尾真，辻井潤一：日本語分析資料及びツールの調査，情報処理，20 (10) PP，941-947 (1979)
2）植村俊亮：電子計算機による自動索引の研究（上・下），電子技術総合研究所報告 743,747 (1974)
3）国立国語研究所：電子計算機による国語研究Ⅰ～Ⅹ
4）田嶋一夫，星野雅英：本文研究の立場からみたテキスト・データの機能について，情報処理学会自然言語研究会資料32-6 (1982)
5）田嶋一夫，星野雅英：国文学語彙検索システム及び索引誌の作成に関する研究報告書(代表：市古貞次)，66 P (1982)
6）国文学研究資料館：漢字データ処理用ソフトウエア，国文学研究資料館報告3号 (1978)