数理解析研究所講究録
第423巻 1981年 2-19

2

文献集作成補助システムの設計と開発

京都大学　工学部　　上林弥彦
Le Viet Chung
徳田成穂
矢島脩三

1. まえがき

学問や技術の高度化、広範囲化、分化、学際化等により、Technology Transfer上の問題として知られている研究成果の流通部分のネックがますます大きな問題となりつつある。この問題を解決する一つの手段として、利用し易い学問、技術に関するデータベースの開発がある。このようなデータベースには次の二種があり、互いに補い合う性格を持っている。

(1) 大量のデータを機械的、網羅的に収集した大規模データベース

(2) 特定の分野の専門家が、個人あるいはグループで収集した詳しく吟味されたデータベース

(1)は、商用データベースや国際的な機関が中心になり分担

して収集されるようなものにみられ、(2)は、非常に限られた分野について各所で作られているようである。(2)の場合、特に良質のデータベースを作るためには、作成者自身がその分野の有能な研究者であることが要求されるが、そのような研究者にはプログラムを開発したりデータを入力する時間的余裕があまりないという問題がある。本稿は、このような問題点を解決するために、比較的容易なデータベース作成のために開発しているシステムについて述べたものである。

我々の研究グループは、文部省特定研究"情報システムの形成過程と学術情報の組織化"(昭51~53年度)の開始と共に、北川敏男先生の御指導のもとに、文献データベース関係の文献の収集を行ってきた。特定研究終了時に1500件であった文献データ数も現在3900件となっている。これらの文献を分類(内部的には250分野)して文献集の作成を行った。[1] このためにいくつかのプログラムが開発されたが、効率の良い処理のためには不向きである面もあり、最終版の作成までに多大な時間を費す結果となった。そのときの作業を分析して、文献集を忙しい研究者でも比較的簡単に作れるようにするために設計したのが本稿で述べるシステムである。

文献情報作成のための問題点は、次のようなものが考えられる。

4

(1) データの入力形式　文献には、論文、本、会議録、本の中の論文、会議録の中の論文、レポート、マニュアル等があり、各々について入力すべきデータ項目や形式が異なる。当然ながら、順序集合を扱える機能が必要である。

(2) 不完全なデータの扱い　データ項目のすべてが揃わなくても入力できることに加えて、その値が不明であるか、存在しない値であるか等の判別ができる。

(3) データ項目の一貫性　雑誌や会議録の名称が入力の都度、入力者の覚え違いや省略形の違い等で異なる。

例　CACM, COMUNICATION OF ACM, COMUNICATIONS OF THE ACM

また、NCC (旧SJCC,FJCC) や IEEE FOCS (旧SCTLD,SWAT) のように、会議名が途中で変わることもある。

(4) データの更新及び誤りの訂正　完全に誤りを訂正することは困難であるが、誤りを発見し易くする機能は必要である。

(5) 分類　文献の分類は、著者や標題、出典等、必要とする情報が広い範囲に渡るため、これらの一覧表示を見ながら入力する会話的動作が要求される。

(6) データ作成のための付加情報の扱い　収集範囲の記録、チェック等に役立つ付加情報が追加できる。

(a) 誤りは、できるだけ小さなループでチェックして訂正できるようにする。今まではまとめてチェックしていたので、誤りの発見が大変であった。

(b) 誤りの発見のために各種出力形式が可能であること。（たとえば、著者索引を作ると著者名の誤りがわかる等）このために、各事項によるソートやKWIC等を利用している。

(c) データの内容について、原データと各ステップで使われるデータには形式等の違いがあり、統一的な管理が必要である。

(d) データの信頼性、歴史に関する付加情報を用いる。特に、原論文にあたって調べなおすことが多く、すでにチェックしたものを再チェックしたり、原典の所在が不明であったりしたことがあり、多大の時間を費した。また、データの入力者やチェック状況等の情報も必要である。

(e) データ量が多い時は、データ作成の中心人物がすべての過程に関与することは無理なので、そのような人が必ず関与すべき誤りの訂正と分類の部分から原簿への訂正が特に容易であること。

その他、システム設計上考慮したことは次の通りである。

(a) データベースの共同開発に適していること。

(b) 非手続き的言語で使い易いこと。

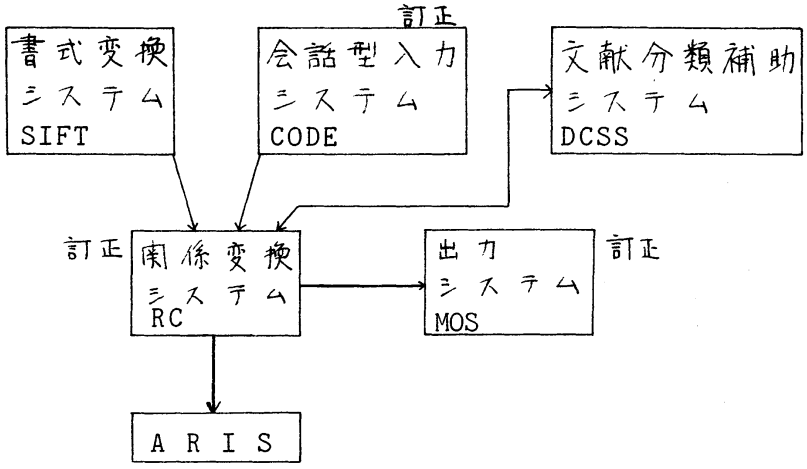
(c) バッチ入力/会話型入力共に可能であること。

(d) 会話的入力の場合は、大型計算機センター TSS のコマンドの用法と似た使い方で特別な使い方を覚える必要のないこと。

(e) システム自体の移植性を考えると共に、我々の研究室で開発しているデータベースシステム ARIS[(2)] とも結合して用いることができる。データベースシステムを拡張してこの機能を持たせると、効率や移植性の点で問題がある。

本稿でデータ入力関係の各サブシステムについて述べ、特に会話型入力サブシステムについて詳述する。

2. システムの構成



本システムは、研究室の関係データベースシステムARISのプリプロセサとして構成され、次の要素より成る。

(1) SIFT (a Simple Form Translator)

(2) CODE (a COnversational Data Entry)

(3) RC (a Relation Converter)

(4) DCSS (a Document Classification Support System)

(5) MOS (a Multi-form Output System)

書式変換システムSIFTは、種々の様式の入力を基本的な関係表に変換するもので、自由度の高い書式を非手続き的に容易に定義できることが特色である。

会話型入力システムCODEは、データベース入力を会話的に行なうためのもので、使い易さを主な目標にしている。

関係変換システムRCは、非正規型(順序集合を含む)の関係を正規化し、編集、訂正するためのものである。

出力システムMOSは、データの一部を特定の属性によってソートしたり、KWICにしたり、総数の情報を出したりするもので、誤りの検出や文献集のための出力に用いられる。

文献分類補助システムDCSSは、KWICとすでに行った分類から、文献分類のための補助情報を出すものである。

エラー訂正は各サブシステムで行なえる形になっており、原データ、非正規化関係、正規化関係等についてもそれぞれ

の訂正が可能で、上位データの訂正は下位データに波及することになる。

3. 文献入力サブシステム

本節では、関係変換システム RC 、書式変換システム SIFT および会話型入力システム CODE について略述する。

3.1 関係変換システム

RC の主な機能は次の通りである。

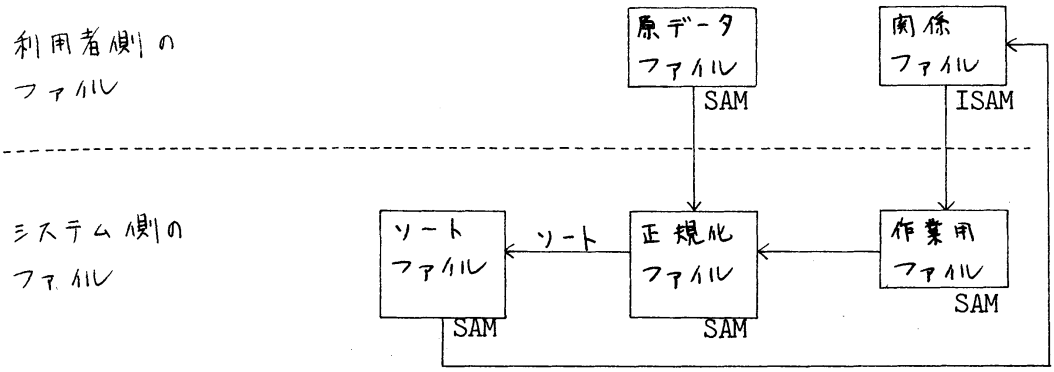
(1) 文献データから関係表を作成する。

(2) 既存関係表を探索したり、追加、削除、変更を行なう。

文献データは非正規的な一次元表現で入力され、対応する関係に関する情報（関係名、属性名、定義域名、属性長（固定／可変）、キー属性、ソート属性）に従って関係表に変換される。 関係表は ARIS の格納形式に対応して ISAM ファイルであり、これに対する変更は ISAM の各ページ単位で行なう。

実際には変更機能は大型計算機センター TSS の EDITOR を利用しているため、図に示すような数種の SAM ファイルが用意されている。 ISAM ファイルはオンラインでアロケートして領域を確保することができないので、新規作成時には前もってバッチ処理で空ファイルを作る必要がある。 他の SAM ファイルについては、オンラインでアロケート／フリーが自由

にできる。



順序集合に対しては、関係表では二つの属性（一つはそのままの属性名、もう一つは順序を示す属性で前者の属性名の後に-0を付けたもの）が対応することになる。

データに対しては次の区切り記号を用いている。

(1) $　属性間

(2) [ ]　属性値の集合　あいまいでない時は省略可

(3) @　繰り返し要素間の区切り記号

(4) /　関係表の中では一連のデータとみなされる区切り記号

@と/は正規化すべきものとしない方が良いものを分けている。システムは部分マッチング機能を持つので、/で区切られたものでも等価的に分離して扱うこともできるが、一般にPP.等のようにその値で検索しない属性に用いられる。

空値には次のものが許されている。

(1) i個存在しているが値は不明　"?i"

(2) 存在しているが数は不明 "?+"

(3) 存在か非存在か不明であるが存在不能ではない "?*"

(4) 非存在 "?0"

(5) 存在不能 "?-"

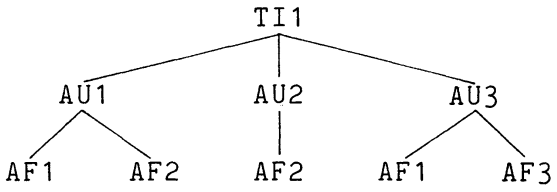
(6) 全く不明 (cr) or "?"

(特に著者一名で et al. の時は Kitagawa/* とする)

入力データに対してトリー展開して対応を決めることになるので、一つの入力データを多くのデータ組に対応させることができる。 たとえば、

TI1$[AU1$[AF1@AF2]@AU2$AF2@AU3$[AF1@AF3]]

は、図に示すトリーに対応する。 このトリーにより順序集合の番号が決められる。



関係ファイルからの読み出しは、一ページごとに行なわれる。 この場合、ページ番号の指定による方法(READ)と、キー値の指定による方法(READ BY KEY)がある。

変更については、次に示すような機能がある。

(1) 属性の追加、属性名やタイプの変更

(2) 関係名の変更

(3) 値域の変更

(4) ソート属性の変更

(5) キー属性の変更

(6) 原データファイルより正規化ファイルへの変更、ソートファイル、関係ファイルへの変換

(7) 原データファイル、正規化ファイル、ソートファイルのエディット（TSS の機能を利用）

(8) STATUS コマンドによる状態の表示

以上の部分については、大型計算機センターで稼動中である。

3.2 書式変換システム SIFT

文献データを見た場合、人間には解釈できるにもかかわらず、機械にかけるためには書式をきっちり定義しなければならないが、これは大変であるので極力避けたい。

P.A.BERNSTEIN AND N.GOODMAN (HARVARD UNIVERSITY AND CCA),
"TIMESTAMP-BASED ALGORITHMS FOR CONCURRENCY CONTROL IN DISTRIBUTED DATA-BASE SYSTEMS", 6-TH VLDB, PP.285-300, OCTOBER 1980, MONTREAL

この例において、所属は（）で、標題は" " で示される。

ページは PP. の後にくる整数又は整数1 - 整数2（整数1 < 整数2）で判別できる。 著者は、C.C*C 又はC.C.C*C の形（C* は0あるいはそれ以上の文字列）で表わされ、OCT. 1980 は値の範囲から日付と判別できる。 このように、値の範囲や形式で

かなり決まってしまい、残る会議名と地名が区別できないが二のような場合にのみ順序による区別を行なう二とにすると、すべての項目が判別できる二とになる。　著者と所属の関係について、BERNSTEINがHARVARDで、GOODMANがCCAである場合には、P.A.BERNSTEIN (HARVARD UNIV.) AND N.GOODMAN (CCA) と書かれるので上の例では直積になる。　二のような自由度の高い書式をいくつも用意しておき、オ一書式でなければオ二書式を適用する----という形式で論文や本を含む我々の書式の処理ができる。　現在、二のための非手続き的な言語と例題による書式規定の機能を持つシステムをSIFTと名付け、開発準備を進めている。

3.3 会話型入力システム CODE

CODEにおいては次のような機能が実現されている。

(1) 会話的入力を行なう。

(2) 柔軟なシステム定義が可能である。

(3) 誤りの検出と訂正を行なう。

(4) データ入力の補助に役立つ情報の管理を行なう。

(5) 文献に限らず、一般的な研究データの会話的入力ができる。

データは属性順に一つずつ入力される。　属性に対して次の区別ができる。

(a) 単一要素よりなる（標題等）か順序集合（著者等）か。

(b) 空値が許されるか。

(c) 初めに指定した値をコピーして後のデータに利用するか。

(d) ほとんど空値であるので必要な時のみプロンプトを出す。 (LANGUAGE,COMMENT etc.)

(e) 一部記号（一般に空白）のサプレス

(f) ある特定の記号が含まれると警告を出す。

また、各データ値に対して特殊な処理をするルーチンを呼ぶことができる。 たとえば、全部大文字のデータを大文字小文字の混ざったデータに変換する、氏名について last name を前に置いたものと後に置いたものとの変換等のルーチンが用意される。

データの誤りの検出と訂正は、次のような機能がある。

(a) 形式のチェック 氏名等 形式に合わないと警告を出す

(b) 値のチェック 年月日等

(c) 簡単な無矛盾性 PP.i-j において i<j か

(d) ID等他の属性から引き出される値のチェック

さらにソート情報を用いてすでに入っている近い値とその頻度とを出すことにより、誤り、同姓同名、二重登録等のチェックをすることも考えている。 さらに MULTI-KWIC システム

と同様のデータ圧縮を行なうと、これらの操作が簡単に実現できると共に、同じデータの有無のチェックも容易である。

新しいデータ値を入れる時、データ形式をそろえるように注意すべきもの（著者名、雑誌名）とあまり関係のないもの（標題等）とに分けられる。

データ入力補助情報として次のようなものがある。

(1) 原典の所在地

(2) データの状態　　何時、誰がチェックしたか

(3) データの信頼性　　直接原典を見て入力したか

(4) 各属性値に対する注釈　　個人の入力データに対して、一時的、要再チェック等の記述

(5) その他の注釈　　to appear、レポートと論文の関係等

これらは、データ管理の難しさを緩和するために必要である。　属性の入力順は自由に定義できる。　また、次章で述べるように、重複して入力することをできるだけ避けるため暗黙指定値を次の各段階で定義できる。

(1) システム起動時にカタログファイルを参照

(2) セッション開設時に(1)を補う形で定義

(3) セッション中に一時的に定義

(4) 直前に入力されたデータの項目値を繰り返す（copy）

4. 会話型入力システムのユーザインタフェース

CODE システムは、オンライン用途のみに限定して設計、開発されているので、プロンプティング及びメッセージコントロール機能が充実している。設計に際しては、大型計算機センターの TSS システムを参考にして便利な機能は共通して用いるだけでなく、独自の必要な機能も加わっている。

以下に各機能について述べる。

先ず、端末への表示はユーザの使用感を左右する重要な要因であり、充実した機能が望まれる。

(1) 各モード表示を各ステップ毎に行なう。

(2) メッセージのヘッダラベルの表示の有無を選択できる。

ヘッダラベルには、メッセージ番号と重要度識別子が含まれており、マニュアルと対応する。

例. CODE1204E ILLEGAL COMMAND SYNTAX

この部分を消すことができる

I --- 通知のみ
W --- 警告
E --- 誤入力
A --- 入力待ち

(3) 初心者向けと熟練者向けのメッセージを選択できる。

何度も使っているユーザには分かり切ったメッセージの抑制ができる。

(4) 空値入力に対して、空値を許す項目と許さない項目は

メッセージの末尾の"-"で判別し、かつ許さない項目に対する空値入力に対しては再入力を促す。

例、CODE1002A MAY I HAVE YOUR NAME ? -
(cr)
CODE9001A REENTER -

(5) ディスプレイ方式とプリンタ方式の端末の特性に合わせてスキップ制御ができる。これは、プロンプトを出力した後、改行するかどうかを選択するものである。

例、CODE0001A READY
NOHEAD
READY
DATE
02/18,1981
READY
NOSKIP
READY TIME
10:45:30
READY (cr)
READY

下線はユーザからの入力

(6) 入力行の最後に"+"を入れて改行すると、テキストがまだ続くものとみなされ次の入力行と"+"を除いて接続される。ただし"+"が正当の入力としてテキストの最後にある場合、"++"と重ねると"+"一文字とみなされ、入力は終了する。

(7) コマンドを";"で区切り、並べて入力するとそのコマンド群を左から順に処理する。

例、DATE;TIME;RESET;INPUT

(8) コマンドには適当な省略形が用意される。

例. I = INPUT

V = VERIFY

(9) 集合入力に対しては、項目名を表示した後、メンバの入力を促す。 集合入力の終わりは空行を入力する。

例. AUTHOR -
+ (cr)
+ YAJIMA,S.
+ KAMBAYASHI,Y.
+ (cr)
TITLE -

(10) 集合入力の区切り記号は内部で自動的に付加されるが、特に指定すれば、特定の文字列を用いることができる。

例. AUTHOR -
+ AU1
+ AU2/AU3 ⟹ AU1/AU2/AU3/AU4
+ AU4
+ (cr)

(11) KWICやソートの出力は適当な大きさのページに分割し、ページの前後へのスキップができる。

本システムでは一つのマスタファイルを複数のユーザが同時にアクセスすることが可能なので、そのための排他的制御が行なわれる。 実際には、マスタファイルに EXCLUSIVE 属性を与え、編集したいレコードが他のユーザによって更新中である場合、そのレコードは現在編集不可であることを表示する。 ユーザはしばらく待って、再びコマンドを入力することになる。

18

INPUTモードの入力制御やチェックは以下の通りである。

これは、できる限り共通のデータを重複して入力することを省くと共に、誤り訂正も容易になる様にするためである。

(1) プロンプトに対して"!"を入力すると、その項目は直前に入力した値を保ったまま、次の項目にスキップする。

(2) "!B"を入力すると、その項目の入力は行なわれず、直前の項目に戻って再入力できる。

例. AUTHOR -
+ KAMBAYASI,Y.
+ (cr)
TITLE -
!B
AUTHOR -
+ KAMBAYASHI,Y.
+ (cr)
TITLE -

(3) 入力の最後にO.K.?と出力された時"!"を入力すると、枠にのみ入力される項目がプロンプトされる。

例. O.K. ?
!
LANGUAGE
FRENCH
COMMENT
ALSO APPEARED CACM 1976
O.K. ?

(4) デフォールト値の扱いはINPUTモードコマンド待ち状態の時にデフォールト項目の指定をすると、最初の入力が指定を解除するまでデフォールト値として登録され、プロンプトも抑制される。一時的にデフォールトを解く場合、"!T"を入力すると抑制されていたプロンプトが出

力され、それに対して値を入力する。　また、デフォールト値を部分的に変更する場合、デフォールト値をセットし直すか、"!D"を入力するとプロンプトが出され、変更された値は以後、デフォールト値として保持される。

また、各項目のデフォールト指定の状態を表示できる。

(5) 各項目について入力時に特定の文字列をサプレスできる。　また、一つ残すか全て消すかの指定もできる。

たとえば、標題、著者においては二つ以上の空白は一つの空白になるが、ページの表現では PP.10-20 のようにすべての空白がサプレスされる。

(6) 特定の文字列が項目中に含まれた時、警告を出力する。

例、DISTRIBUTED DATABASE SYSTEM - SDD-1

これは、KWIC に使う時に"-"や"/"は二つの意味を持っているからである。　最初の一つは、二つの概念をつないでいるが SDD-1 の"-"は単語の中の一文字である。

同様の例は CAM/CAD, PL/I 等があり、この区別をしておかないと良質の KWIC 出力が得られない。

謝辞　名古屋大学プラズマ研の小西修助手、神戸大学教養部の田中克己助手、ならびに研究室の吉川正俊氏、武田浩一氏、小島功氏に感謝する。

文献

(1) Y.Kambayashi(Ed.), O.Konishi, K.Tanaka, C.Le Viet(Ed.Ass.), "Database - A Bibliography Vol.1", Computer Science Press, 1981

(2) S.Yajima, Y.Kambayashi, O.Konishi, K.Tanaka, C.Le Viet and T.Kato, "Bibliographical Information Processing Facilities for Relational Database System ARIS", 13th Hawaii International Conference on System Science, Vol.2, pp.198-207, Jan. 1980