

Lexiconとしての位相情報空間の構成

打浪 清一 (Seiichi Uchinami)
(大阪大学 工学部)

[I] はじめに

意味の包含, 遠近関係を記述するモデルとして, 位相情報空間モデルを提案し, その Formal なモデル化と, その数学的性質について検討してきた。このモデルはデータベース理論の点のデータモデルとしての一面と共に, 自然言語モデルとしての一面も兼ね備えたモデルとして発展させてきた。

自然語処理, 質問応答システム, 知識データベース等とインプリメントする際には, 実際にこの位相情報空間を構成し, これと知識ユニバースとして参照, 利用できるような必要がある。しかしこれを一度に作りあげるのは困難で, 逐次成長させて構成する方が実現性が高い。

そこでこれを構成するために, 次の3つのシステムを構築し, Lexicon構成の準備を行っている。

(1) Conceptual-Scheme-Free DBMS

概念スキーマを創成時に設計しなくても、逐次データと投入していき、内部で自動的にクラスタ化を行って、概念スキーマを構成してゆくシステム。

(2) 知識データベースシステム

知識もデータとして加工処理ができるように、S-M変換可能な記述言語であるLISPを採用し、知識が大量になっても高速にアクセスができるように、ファイル構成を行なったシステム。

(3) 言語知識獲得システム

日本語の簡単なParserをもち、入力文を解析し、短期フレーム、長期フレーム記憶に、各単語の上、下位概念や用法等を整理、学習して蓄積してゆくシステム。

これらのシステムを作成し、実験してゆくうちに、概念の分析整理段階におけるクラスタリング、抽象化とシステムティックに行なえるシステムが必要不可欠であることが判明した。そこで本稿では位相情報空間(意味地図)と、抽象化を行いつながらに構成してゆく方法について検討する。

〔II〕 位相情報空間モデル概説

生成意味論の立場でモデル化している。位相情報空間生成文法が基本規則として定義される。この文法により生成された空間を位相情報空間とよび、この空間が意味地図となる。位相情報空間生成文法から統辞生成文法とよめる写像の存在し、この写像を用いると、各概念領域はそれの表す語、句または文に意味される。

表層の統辞文法では、生成された文の間に何等直接的な意味関係は陽には表われていないので、ただ単に文の集合と生成すればよいが、空間生成文法では点あるいは領域の生成だけでなく、その位相も併せて生成しなければならぬ。ところが使用時には、位相ではなく特定概念領域と指示すればよいから、結局生成規則には空間全体、知識構造と生成するものと、伝達したい個々の概念と生成するものとと二種類必要である。前者をユニバース生成規則、後者を個概念生成規則と呼ぶ。ユニバース生成規則は本モデル特有のものであって、これで位相空間が構成される。他のモデルではこれに該当するものが無いので、知識全宇宙を well-formed に記述できる強力な tool ともたない。

ユニバース生成規則は位相空間全体（構成要素集合と、その上での位相）を生成し、発話の意味解釈時には、概念生成

規則で特定概念領域を生成する。

位相情報空間を構成する各座標軸は、夫々異なる位相を持っている。

位相情報空間は、次のように well-formed に定義される。
位相情報空間生成文法は、生成規則の適用に制限をつけた次の 3 レベルの文法からなる。

[定義 1] 空間生成文法 G_1

$$G_1 = \langle V_{N1}, V_{T1}, P_1, S_1 \rangle \quad \dots (1)$$

$$\text{ここで } V_1 = V_{N1} \cup V_{T1}, \quad V_{N1} \cap V_{T1} = \emptyset \quad \dots (2)$$

V_{N1}, V_{T1}, P_1, S_1 は第 1 レベルの中間語彙, 終端語彙, 生成規則, 初期語彙を表し、他レベルでも (2) に対応する式が成立するとする。

P_1 は次の形の規則からなる。

$$P_1: A \rightarrow \omega \quad A \in V_{N1}, \quad \omega \in L^A(V_1, D) \quad (3)$$

$$L^A(V, D) \stackrel{\text{def}}{=} \{ (\alpha \delta)^* \beta \mid \alpha, \beta \in V, \delta \in D \} \quad \dots (4)$$

$$D = \{ \cdot, \#, | \} \quad \dots (5)$$

G_1 は情報空間を構成する部分空間間の関係を規定するもので、 D は空間構成子で、その種類としては、直積 $\cdot$, 連結和 $\#$, 直和 $|$ がある。

[定義2] 空間性質規定文法 G_2

$$G_2 = \langle V_{N2}, V_{T2}, P_2, S_2 \rangle \quad \dots (6)$$

P_2 は次の形の規則からなる.

$$P_2: A \rightarrow w, \quad A \in V_{N2}, \quad a, b \in V_{T2} \quad \dots (7)$$

$$w \in \{a, \bar{a}, aa, aa^{-1}, aba^{-1}b^{-1}, aba^{-1}\} \quad \dots (8)$$

G_2 は各部分空間の性質を規定するもので、その性質は w の要素で定まる。その種類には、

(i) 直線軸 (a)、 $\mathbb{R}$, $\mathbb{N}$, $\mathbb{Z}$ 、またはその閉、開区間、

(ii) 円 ($\bar{a}$)、(i) の直線軸の両端点を同一視したものの。

以下位相幾何学の多様体を表す多角形式の

(iii) 射影平面 (aa)

(iv) 球面 (aa^{-1})

(v) 一定の輪環面 ($aba^{-1}b^{-1}$)、トーラス

(vi) 一定面 (aba^{-1})

等の制約型がある。

このレベルで座標軸を組み合わせて部分空間を構成する方法と定める。

[定義3] 位相規定文法 G_3

$$G_3 = \langle V_{N3}, V_{T3}, P_3, S_3 \rangle \quad \dots (9)$$

P_3 は知識全宇宙を生成する P_{3T} と、空間内の領域を生成する P_{3A} とからなる。

$$P_{3T} : A \rightarrow \omega, \quad \omega \in \{\text{各型、位相空間}\} \quad \dots (10)$$

$$P_{3A} : A \rightarrow \omega, \quad \omega \in \{\text{位相空間の領域}\} \quad \dots (11)$$

G_3 は部分空間の位相を与えるもので、その位相の型としては、次のようなものがある。

$T_0, T_1, T_2, R, T_3, CR, T, N, T_4, CN, T_5, FN, PN, T_v, L, C, CH, LC, C_n, U, S, QM, M, B, \dots$

以上の3レベルを統合して 位相情報空間 が生成される。

[定義4] 位相情報空間 生成文法 G_I

$$G_I = \langle G_1, G_2, G_3 \rangle \quad \dots (12)$$

ここで 3レベルの間の関係は次のように定められる。

$$S_1 = S, \quad S_2 = V_{T1}, \quad S_3 = V_{T2} \quad \dots (13)$$

各レベルの適用規則に制限をつけることにより、種々の空間が構成される。

[定義5] 位相情報空間 I

$$I = I(G_I) \quad \dots (14)$$

位相情報空間生成文法 G_I から生成された位相空間 I を、位相情報空間という。このとき第3レベルでは P_{3T} のルール

を用いる。この空間が言語モデルにおいては、Lexicon と構成する。データベースにおいては、概念スキーマに対応し G_E のこのスキーマ記述に用いられる。

〔Ⅲ〕 位相情報空間の構成法

Lexicon としての位相情報空間を、全データを集めて一括して構成することは、ユニバースの点から考えて実現し難い。それ故ローカルな分野毎にデータエントリを行い、ローカルな位相情報空間を逐次構成し、その後これらを統合してよりグローバルな位相情報空間を構成してゆくことになる。

この段階において、データの整理、分類、抽象化、構造化が行なわれる。

位相情報空間型データベースシステムに於ては、最もカリミティブな要素として属性があり、その上に個体、集合、線、領域、... と、更にこれらの再帰的に入ったものとしてデータタイプが定義される。

関係モデルは plane の一レベルしかなく、個体関係性モデルでは二レベルであるが、本モデルは多レベルであり、各レベルでそのタイプの持つトポロジーで幾つかの種類がある。関係性は本モデルでは集合や線、領域に該当するものと

が多い。

位相情報空間の定義は、位相情報空間型DBMSのDDLによってなされる。ローカルな構成情報が入力され、それが統合され、概念スキーマに相当するLexiconの構成が定義されてゆく。このために次のような入力が行われる。

各軸の詳細 property specification

空間の構成法 Space coordinate specification

空間と空間の統合法 Space composition specification

各軸の定義においては、次のような属性が定義される。

属性名, Topology Type, Data Type, Range,

Occurrence, Ordering, 単位 etc.

ここで Topology Type とは、(8), (10)に関連したタイプで、数軸とか、循環数軸、順序軸、...等の区別をいい、Data Type とは 整数型、実数型、文字型...等の区別をいう。

空間の構成法においては、空間構成の定義が行われる。

空間名, 空間構成軸とその構成法, Space Data Type etc.

ここで Space Data Type とは、個体, 集合, 領域...etcの区別をいう。

空間と空間の統合法においては、空間の合成法が定義され

る。

空間名, 部分空間名, 空間の合成法. Space Data Type etc.

空間の構成は Recursiveに行われる。即ち低位のレベルでは、空間を構成する要素として属はか入るが、次第に上位のレベルにゆくに従って、個体や部分空間と要素として空間が定義される。

このようにローカルに定義されたデータと、空間生成文法の生成規則の形でデータベースに蓄積してゆく。

これとともにグローバルな空間構成を求め子には、抽象化技法を援用する。抽象化には、データの抽象化、手続きの抽象化、制御の抽象化がある。位相情報空間の整理統合に用いるのは、主にデータの抽象化であるが、データの抽象化と通して、手続き、制御の抽象化とも組っている。これらは、手順知識の抽象化や DMLの抽象化となって効いてくる。

データの抽象化の効用は、知識ユニバース形成の後の概念整理、概念スキーマ作成の後のシステム分析、知識データ構造の拡張性等がある。

その抽象化レベルを図1に示す。図1に於て ext. relationship level は、個体を構成要素として、集合や領域、線、...と構成するレベルを一括して表したものである。

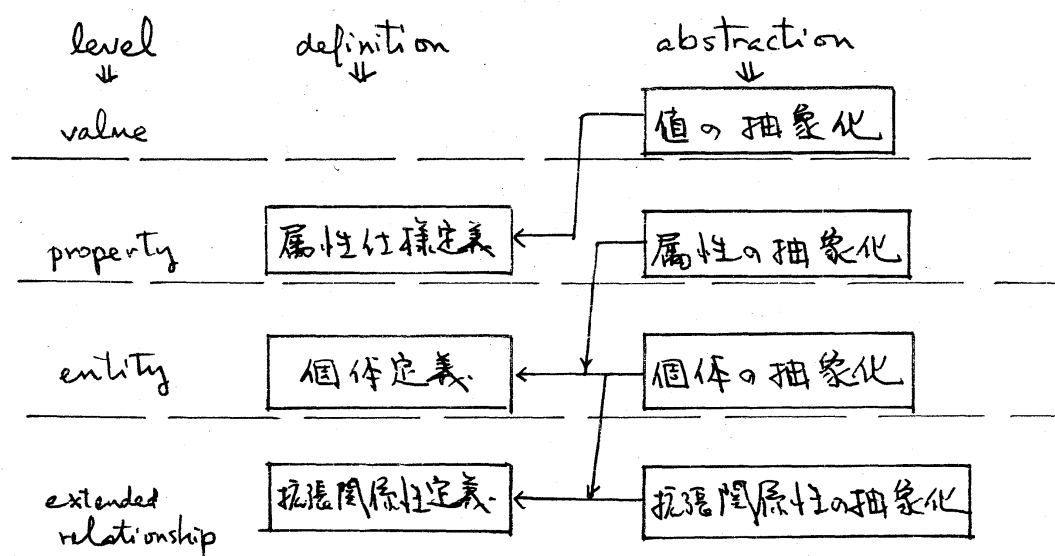


図1. 位相情報空間型DBSにおける抽象化レベル

定義は Recursiveに行われるので、定義された関係、例えば集合が、もう一段上の抽象化されたレベルでは、個体として扱われることもあり得る。点線は Aspect の変更による level としての role の変更を表している。

[IV] 位相情報空間の抽象化法

概念の抽象化には種々のタイプが考えられる。知識ユニバースを合成してゆくに際しては、できるだけ種々の抽象化が可能のように定義するのが望ましい。

先ず抽象化のタイプを挙げ、次にそのお理法と述べる。

(1) クラスの包含関係による抽象化: a kind of (AKO)

A is a kind of B

$A \neq B$ instance ではなく集合であるクラスを表し,
 $A \subset B$ である。

(2) クラスとインスタンスの関係による抽象化 : is a

A is a B specialization

A は B の instance であり $A \in B$ である。

(3) 個体の集合か別の個体を形成する抽象化 : member of

A is association of B members.

$B \sim$ member を連合して、一つの組織 A を構成している。

(4) 部分全体関係による抽象化 : a part of, aggregation

A is a part of B

A は B を構成する要素である。

(5) 構成要素と個体の関係による抽象化 : is made of

A is made of B

A は B と組み合わせて構成されている。

(6) 構成原料と個体の関係による抽象化 : is made from

A is made from B

A は B を原材料として構成されている。

(7) 性質として持つという関係にある抽象化 : has

A has B

A は性質 B を持っている。

ここに挙げたものは必ずしも素ではないが、概念間の関係を記述したい際にあらわれてくる抽象化に関連したものである。

また抽象化とは、乏しいかもしれないが、次のような性質がある。

(1) Weak, Strong property

Weak: その性質と持っても持たなくてもよい。

Strong: 必ずその性質を持つ。

(2) Stable, Not stable property

Stable: 一度その性質と持てばずっと変らないもの。

Not stable: その性質と持っても、また持たなくなることもあり、変化するもの。

(3) 共生, 排反関係

共生関係: $(A \rightarrow B) \cap (A \rightarrow C)$ B, C が両方同時に起こる。感いは持つ。

排反関係: $(A \rightarrow B) \cap (A \rightarrow C) = \emptyset$. 両方同時に起こらない。

(4) 直積関係

直積関係: $A \rightarrow B \cdot C$ B, C の直積として定義される。

これらのローカルな情報と生成規則の形で投入し、それと整理して、定義4の位相情報空間生成文法を作成する。未整理生成規則の投入の他に、生成規則を記述するメタ・ルール
の他に、これら各種の抽象化の各の情報と表現するメタ記号を導入する。

⊃ part AKO abstraction $A \supset B$: B は A の下位クラスである。

⊇ full AKO abstraction $A \supset B \cup C$: B と C は A の下位クラスでこれらで覆い尽くされる。

⇒ part is-a abstraction $A \Rightarrow B$: B は A の instance である。

⊇ full is-a abstraction $A \ni \{B_i\}$: B_i は instance で $\{B_i\}$ が A を構成する。

→ part association abstraction $A \rightarrow B$: 個体 B の集合が A を構成する。

⇒ full association abstraction $A \Rightarrow B$: 個体 B の集合で A 全体が構成される。

↓ part (part) aggregation $A \downarrow B$: BはAの構成要素
の一つ。

↓ full aggregation $A \downarrow \{B\}$: $\{B\}$ でAは構成さ
れる。

$\overset{of}{\downarrow}$ part is-made-of abstraction $A \overset{of}{\downarrow} B$: BはAの構成要素
の一部

$\overset{of}{\downarrow\downarrow}$ full is-made-of abstraction $A \overset{of}{\downarrow\downarrow} B$: $\{B\}$ はAの構成要素

$\overset{from}{\downarrow}$ part is-made-from abstraction $A \overset{from}{\downarrow} B$: BはAの原料の一部

$\overset{from}{\downarrow\downarrow}$ full is-made-from abstraction $A \overset{from}{\downarrow\downarrow} B$: $\{B\}$ はAの原料の全部

$\overset{has}{\downarrow}$ part has abstraction $A \overset{has}{\downarrow} B$: BはAの性質の一部

$\overset{has}{\downarrow\downarrow}$ full has abstraction $A \overset{has}{\downarrow\downarrow} \{B\}$: $\{B\}$ はAの持つ性質
の総て。

概念の包含, 遠近, 階層関係等を整理したとき, 各概念の
持つ性質の下位, 上位, 上位概念への遺伝性の問題になる。
性質の遺伝のタイプには, 次のようなものがある。

(1) 全部が伝わる。

(2) 一部が伝わる。

(3) 伝わりない。

これらの類別も上記記述から導出できるものの分析中である。逆にいえば、 $\downarrow$ の各種別と、伝わるもの、一部が遺伝するもの、遺伝しないものというように分けることができれば処理しやすいと考えられる。

例えば $A \supset B \Rightarrow b$ は、 A の性質が B に遺伝し、 b はその性質を持つ。

これらのメタ記号を用いて記述され、投入された位相情報空間構成情報は整理され、生成規則へと抽象化される。

空間の構造の抽象化において、次の補題が成立する。

[補題1] aggregation 分配律

$$A \downarrow (B|C), A \downarrow (D|E) \Rightarrow A \downarrow (B|C) \cdot (D|E) \quad \dots (15)$$

$$A \downarrow B, A \downarrow C \Rightarrow A \downarrow (B|C) \quad \dots (16)$$

(15)は、 B, C が A に full aggregate すると同時に、 D, E が A に full aggregate すれば、 A は $(B|C) \cdot (D|E)$ の直積の形に構成されることを示す。

$$\left. \begin{array}{l} \text{Ex. } \text{人} \downarrow (\text{肉体}|\text{精神}) \\ \text{人} \downarrow (\text{姓名}|\text{年齢}) \end{array} \right\} \Rightarrow \text{人} \downarrow (\text{肉体}|\text{精神}) \cdot (\text{姓名} \cdot \text{年齢})$$

(16)は、 B, C が A に partially aggregate するとき。

$A \downarrow (B|C)$ と処理して良いことを示す。

ex. 人 $\downarrow$ 肉体, 人 $\downarrow$ 精神 $\Rightarrow$ 人 $\downarrow$ (肉体|精神)

人 $\downarrow$ 肉体, 人 $\downarrow$ 姓名 $\Rightarrow$ 人 $\downarrow$ (肉体|姓名)

part aggregate は必ず統合しても full aggregate は必ず出せない。また OR で結んでゆくものは同値のものとは限らないので、おかしい aggregate も起き得る。

[補題2] aggregation 吸収律

$$A \downarrow (B|C), A \downarrow (B|C|D) \Rightarrow A \downarrow (B|C|\widehat{D}) \quad \dots (17)$$

$$\text{ここで } \widehat{D} = (D|\Omega)$$

$$A \downarrow (B|C), A \downarrow (B|C|D) \Rightarrow A \downarrow (B|C|D) \quad \dots (18)$$

(17)は full aggregation において、値域の一部重複、一部異なる形で定義されている場合の吸収律を表す。この場合(17)の左辺の2つの式が同時に成立することはあり得ないことを示すこともできるが、そのデータエントリの際に十分考えておくのではなく $\downarrow$ を選んだのだから、やはりそれには存在する場合があるのだと考える。それ故 $A \downarrow (B|C)$ より、ある場合には B, C で full aggregate されておき、またある場合には $A \downarrow (B|C|D)$ より B, C, D で full aggregate されているとみなし、そのため D はある場合には null 値 Ω であると判定し (17)のように統合される。

(18)は含まれる方の part aggregate の場合で、この場合は完全に吸収される。

[補題3] aggregation 統合律

$$A \downarrow (B|C), A \downarrow (C|D) \Rightarrow A \downarrow (B|C|D) \quad \dots (19)$$

$$A \downarrow (B|C), A \downarrow (C|D) \Rightarrow A \downarrow (\bar{B}|C|\bar{D}) \quad \dots (20)$$

$$\bar{B} = (B|\Omega), \bar{D} = (D|\Omega)$$

空間を構成する座標軸が部分的に与えられてそれと合成する場合の統合律である。

[補題4] AKO 分配律

$$A \supseteq (B|C), A \supseteq (D|E) \Rightarrow A \supseteq (B|C) \cdot (D|E) \quad \dots (21)$$

$$A \supset B, A \supset C \Rightarrow A \supset (B|C) \quad \dots (22)$$

Ex. $人 \supseteq (男|女), 人 \supseteq (子供|大人) \Rightarrow 人 \supseteq (男|女) \cdot (子供|大人)$

Ex. $人 \supset 男, 人 \supset 子供 \Rightarrow 人 \supset (男|子供)$

$人 \supset 男, 人 \supset 女 \Rightarrow 人 \supset (男|女)$

この場合も観点が異なるものをまとめると意味的におかしい場合も生じる。

[補題5] AKO 吸収律

$$A \supseteq (B|C), A \supseteq (B|C|D) \Rightarrow A \supseteq (B|C|\bar{D}) \quad (23)$$

$$\bar{D} = (D | \Omega)$$

$$A \supset (B | C), A \supset (B | C | D) \Rightarrow A \supset (B | C | D) \dots (24)$$

[補題6] AKO統合律

$$A \supset (B | C), A \supset (C | D) \Rightarrow A \supset (B | C | D) \dots (25)$$

$$A \supseteq (B | C), A \supseteq (C | D) \Rightarrow A \supseteq (\bar{B} | C | \bar{D}) \dots (26)$$

同様に $\exists, \Rightarrow, \overset{of}{\downarrow}, \overset{from}{\downarrow}, \overset{has}{\downarrow}$ や $\ni, \rightarrow, \overset{of}{\downarrow}, \overset{from}{\downarrow}, \overset{has}{\downarrow}$ についても種々の律が成立し、またこれらと混合した律も成立するか以下省略する。

[V] 情報空間ファイルの構成

前述の抽象化法により逐次構成されてゆく P_{ST} は、知識ユニバーサスの概念スキーマを形成する。

ENTITY NAME	ID. NO.	FACET , i,	NUMBER OF DATA	GENERATIVE RULE NO.S IN DERIV. TREE	NUMBER OF DATA	VOCABULARIES IN DERIVATION TREE
Season	10	,All,	8	3, .. 2, .. 1	25	3,..1
Spring	11	,C,	9	5,3,..2,..1	26	5,3,..1
Summer	12	,D,	9	6,3,..2,..1	26	6,3,..1
Autumn	13	,E,	9	7,3,..2,..1	26	7,3,..1
Winter	14	,F,	9	8,3,..2,..1	26	8,3,..1
January	142	,F,	10	9,8,3,2,..1	27	9,8,3,..1
Feb.	143	,F,	10	10,8,3,2,..1	27	10,8,3,..1
March	111	,C,	10	11,5,3,2,..1	27	11,5,3,..1
April	112	,C,	10	12,5,3,2,..1	27	12,5,3,..1
May	113	,C,	10	13,5,3,2,..1	27	13,5,3,..1
June	121	,D,	10	14,6,3,2,..1	27	14,6,3,..1
July	122	,D,	10	15,6,3,2,..1	27	15,6,3,..1
..				..		

$\uparrow$ $\uparrow$ $\uparrow$ $\uparrow$
 使用 生成規則 語法 語彙番号
 規則表 番号

図2 ソースデータファイル

GENERATIVE RULES		VOCABULARY		
NO.	RULE	NO	V	$\phi(V)$
1	$S \rightarrow A_1 A_2 A_3$	1	S	..
2	$A_1 \rightarrow [B_1 B_2 B_3]$	2	A_1	..
3	$A_2 \rightarrow [C D E F]$	3	A_2	season
4	$A_3 \rightarrow [V_1 V_2 V_3]$	4	A_3	...
5	$C \rightarrow [G H I]$	5	C	spring
6	$D \rightarrow [J K L]$	6	D	summer
7	$E \rightarrow [M N O]$	7	E	autumn
8	$F \rightarrow [P Q R]$	8	F	winter
9	$G \rightarrow \dots$	9	G	January
		10	H	February
		11	I	March
		12	J	April
				...

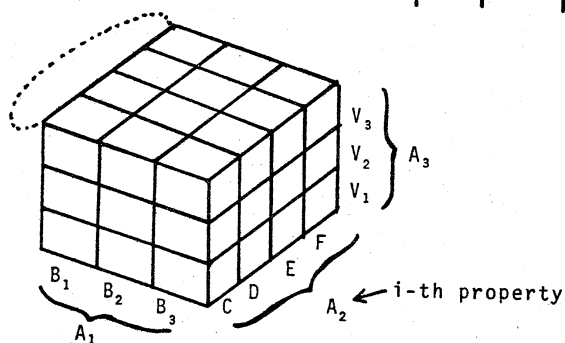


図3 空間生成規則と空間の構成例

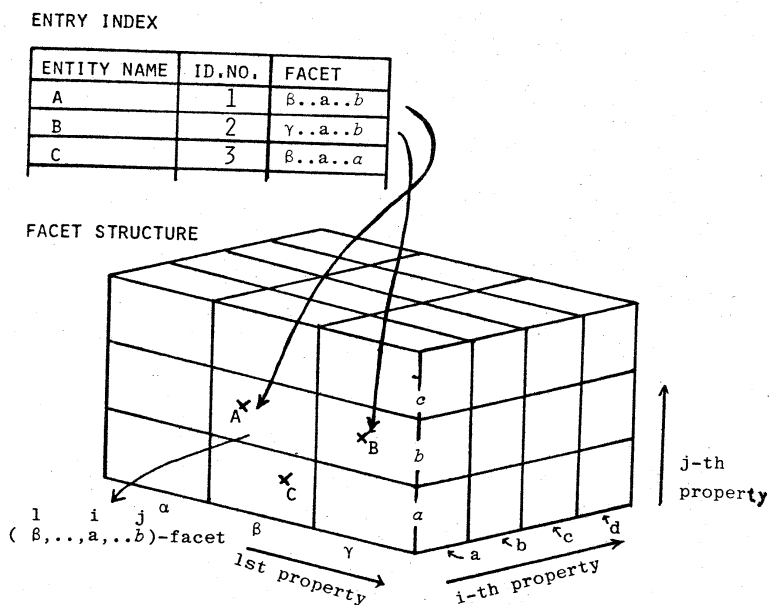


図4 ファイル構成図例

INVERTED PROPERTY INDEX					
1ST PROPERTY		I-TH PROPERTY		J-TH PROPERTY	
a	A, C	a	A, B, C	a	C
B	B	b		b	A, B
Y		c		c	
		d			

SOURCE DATA FILE					
	P_1/U_1	P_2/U_2	P_3/U_3	P_4/U_4	...
ENTITY NAME	ID. NO.	PROPERTY-1	..	PROPERTY-I	PROPERTY-J
A	1	B		a	b
B	2	Y		a	b
C	3	B		a	a

図5 転置索引と一次ファイルの構成例

生成規則から位相情報空間と生成する構成法を図2, 及び図3に示す。位相情報空間型データベース構成法を図4及び図5に示す。

[Ⅵ] むすび

総ての観点からの知識を網羅した意味地図ともいうべき位相情報空間を構成してゆくには、ローカルな空間情報と、生成規則の形で投入し、抽象化し、階層構造及び位相関係を構築してゆく必要がある。その為のデータの入力法とその抽象化整理法について分析した。現在データ投入の為のソフトウェアのインプリメンテーションと並行して、抽象化の為のデータベースの整備を行っている。

文献

- (1) 打浪他「位相情報空間生成文法とその空間の性質」電子通信学会論文誌'80年1月号 J63D1
- (2) 土井, 打浪, 千塚「概念スキーマフリー-DBMSの設計とそのデータ抽象化法について」. 信学会オートマツンと言語研究'83年1月
- (3) 楠野, 打浪, 千塚「日本語の意味構造と表現する中間言語について」信学会オートマツンと言語研究'83年3月