知能と複雑系 116－2
（1999．5．24）

構造化文書の論理的扱い

有馬 淳
（株）富士通研究所
〒814-8588 福岡市早良区百道浜 2-2-1
E-mail: arima@flab.fujitsu.co.jp

あらまし

マークアップ言語によって構造化された文書を論理的な方向で捉え直す試みを行なっている。本論文では、その初期考察結果を報告する。RDB や Web の表形式で表された知識や、XML などによって構造文書の論理構造を表現する共通表現系とその推論系を提案する。また、線形論理によって推論系の形式的意味を与える。

Logical Treatment of Structured Document

Jun ARIMA
Fujitsu Laboratories LTD.
2-2-1 Momochihama, Sawara-ku, Fukuoka 814-8588, Japan
E-mail: arima@flab.fujitsu.co.jp

Abstract

This paper describes an attempt to treat structured document with mark-up language in a logical manner. Introducing a unified representation system for knowledge in table form of HTML and in relational data-base, and tagged document with XML and etc, we propose an inference system for the system。 A semantic meaning of the logical system is given using by linear logic.

1 はじめに

7、80年代、人工知能の工学的な応用を阻む状況の一つに、人工知能システムが扱うべき「知識」そのものが電子媒体上になく、入力もまた現実的ではないという状態があった。これは「知識獲得問題」と呼ばれる重要問題の少なくとも一部を形成していた。しかしながら、近年のインターネットの普及、技術の進展に伴いこの状況は大きく変わりつつある。特に注目すべきは、HTMLに代表されるマークアップ言語の普及であり、さらにXMLの登場によって表示用のタグから意味的なタグへと移行することで、上記の状況は大きく改善される可能性が高まってきた。現在のところ構造化文書の扱いはシンタックス中心であるが、「知識」源としてセマンティクスを重視し、両者を融合する扱いを目指したい。

本論文は、このような観点からマークアップ言語によって構造化された文書を論理的な方向で捉え直す試みの初期考察結果を報告している。本稿では、特にRDBやHTML表形式の知識や、XMLなどタグつき文書の論理構造を表現する共通表現系とその推論系を考察する。

2 課題領域

2.1 表形式（RDB）

図 1の表を例にとり、機械的に表形式を述語表現する2つの翻訳方法をあげ検討する。

形式 A:

搬入DB(6921, 597, ノート) ∧ 搬入DB(6928, 595, ボールペン)
∧搬入DB(6951, 587, 鉛筆) ∧ 搬入DB(6940, 597, ボールペン)

形式 B:

伝票ID($t1$, 6921) ∧ 取引先ID($t1$, 597) ∧ 品名($t1$, ノート)
∧伝票ID($t2$, 6928) ∧ 取引先ID($t2$, 595) ∧ 品名($t2$, ボールペン)
∧伝票ID($t3$, 6940) ∧ 取引先ID($t3$, 597) ∧ 品名($t3$, ボールペン)
∧伝票ID($t4$, 6951) ∧ 取引先ID($t4$, 587) ∧ 品名($t4$, 鉛筆)

形式Aは最も代表的な翻訳の一つである[1]が、欠点としては、属性情報が陽に表示されず、引数位置に暗黙的に与えられていることがあげられる。形式Bは、各タプルを t_i で表し属性を述語名で表したものであるが、1) $t1, t2$ など元の表に現れない構造上の情報が表現に出る。また2) join操作によって得られるタプルの意味をとると、以下のようになり

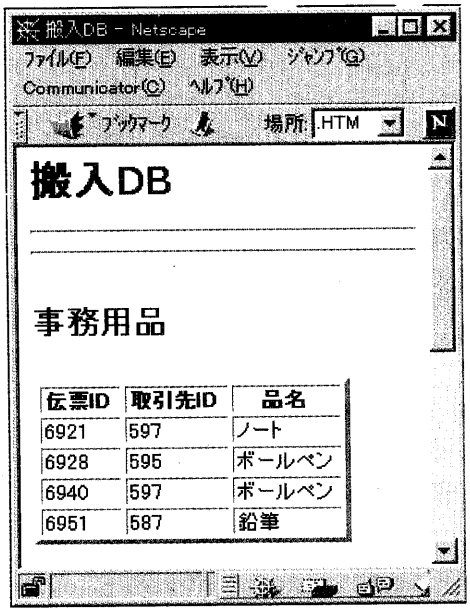

図 1: 搬入 DB （HTML 形式）

形式 Aが Horn 文で表されるのに比べ操作に特殊な扱いが必要になる。

品名($D1, X$) ∧ 取引先ID($D1, I$) ∧ ID($D2, I$) ∧ 社名($D2, N$)
⊃ ∃T(品名(T, X) ∧ 社名(T, N))

2.2 XML

XML文書要素は以下のような木構造を持つ[2]。

文書要素 :=
　文書 | < 要素名 属性指定/ > |
　< 要素名 属性指定 > 文書要素の集合 < /要素名 >

図 2のXML文書例を考える。本例はまた図 3のように図示される。このようなXML文書を述語表現にする2例を以下に記す。
形式 C:

has_a(取引先DB, 取引先) ∧ has_a(取引先, 社名)
∧ID(取引先, 597) ∧ is_a(社名, 春日商店)
∧ID(取引先, 595) ∧ is_a(社名, 山田文具)
∧ID(取引先, 587) ∧ is_a(社名, 三省社)。

```
<取引先DB>
  <取引先 ID="597">
        <社名> 春日商店 </社名>
  </取引先>
  <取引先 ID="595">
        <社名> 山田文具 </社名>
  </取引先>
  <取引先 ID="587">
        <社名> 三省社 </社名>
  </取引先>
</取引先DB>
```

図 2: 取引先 DB （XML 形式）

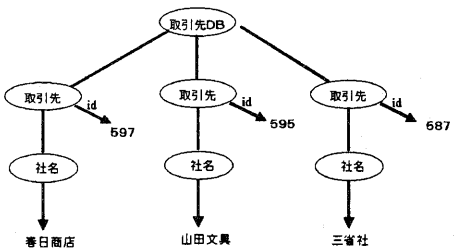

図 3: XML tree

形式 D:

$$branches(n, n1) \wedge branches(n1, n11)$$
$$\wedge branches(n, n2) \wedge branches(n2, n21)$$
$$\wedge branches(n, n3) \wedge branches(n3, n31)$$
$$\wedge label(n, \text{取引先}DB)$$
$$\wedge label(n1, \text{取引先}) \wedge label(n11, \text{社名})$$
$$\wedge label(n2, \text{取引先}) \wedge label(n21, \text{社名})$$
$$\wedge label(n3, \text{取引先}) \wedge label(n31, \text{社名})$$
$$\wedge attrib(n1, ID, 597) \wedge is\_a(n11, \text{春日商店})$$
$$\wedge attrib(n2, ID, 595) \wedge is\_a(n21, \text{山田文具})$$
$$\wedge attrib(n3, ID, 587) \wedge is\_a(n31, \text{三省社})$$

形式 C は論理積の性質から ID 番号の 597 と社名の春日商店の間の対応関係が落ちてしまう点で明らかに不適切な表現である。

形式 D では、各XML文書要素に一意の項 $n, n1, \cdots$ を付与し、その項との関係で表現した。元の情報を落さないが、形式 B と同様元の表現に現れない構造上の項が出てくる点、また、元の情報に比べ読みにくい点などが不満足である。

3 構造化述語と推論

柔軟な意味処理のための論理系を提案する。
定式化の基本的な考え:

i) 表は木構造で表せる（例：表をトップノード、各タプルを中間ノード、属性値をリーフ）。従って木構造をうまく表現できれば良い。

ii) 木構造を論理式で表現すると (形式 D のように)、内容に関するデータと構造に関するデータが混ざり、全体が読みにくい。また、木構造特有のデータ操作が多いと予想されるが、一般的な操作では実現効率が悪い。

そこで、木構造をそのまま基本項とする構造化述語を導入し、それを処理、意味を与える論理系を与える。

なお、本論文では木構造の扱いのみに注目するため、XML の特徴の一つである文書要素間の順序は扱わない。また、論理性に焦点をおくため属性指定は文書要素と同一視する[1]。

3.1 導入

述語から構造化述語への直観的な流れを順に示す。

述語： 引数の位置に過敏で、その位置の意味は暗黙的。知識の流通には問題がある。

$$p(a, b, c)$$

ラベルの導入： 引数位置／型を表すラベルを各引数に表示。引数位置を鈍感にする。

$$p(l\#a, m\#b, n\#c) \equiv p(n\#c, l\#a, m\#b)$$

引数の可変化： 引数の省略を関係づける。

$$p(l\#a, m\#b, n\#c) \supset p\{l\#a, n\#c\}$$

構造化： 述語、項の関係を見直し、すべてラベルと見、木構造へ一般化する。

$$p\#\{l\#a, m\#b, n\#c\},$$
$$q\#\{\ p\#\{l\#a, m\#b, n\#c\}, p\#\{l\#d, m\#e\}\ \}$$

[1] すなわち、取引先 DB の例では ID=''597'' は <ID> ''597'' </ID> と同じ扱いをする。もし要素と区別する必要があれば <attrb:ID> ''597'' </attrb:ID> など属性を表すタグ名にする方法も考えられる。

3.2 定義

構造化述語 (*structured predicate*)、**内容** (*content*)、ラベル (*label*)、**項** (*term*) のシンタックスを以下に与える。

定義 1

$$
\begin{array}{lcl}
\textit{structured predicate} & := & \textit{label}\#\textit{content} \\
\textit{content} & := & \epsilon \mid \text{set of } \{\textit{predicate}(s) \mid \textit{term}(s)\} \\
\textit{label} & := & \textit{term} \\
\textit{term} & := & \textit{alphabets}
\end{array}
$$

ここで ϵ は空集合を表す。 □

意図する対応づけ：

structured predicate	↔	文書要素
content	↔	文書内容
label	↔	タグ(や属性名)
term	↔	文字列

label はその構成要素に対する述語であり、（HTML のような書式タグでなく）XML での意味タグや RDB での属性名に対応させる意図がある。それゆえ、$l\#\mathcal{P}$ は文書要素 $\mathcal{P}$ が l であることを示す。また、$l\#\{p_1,\cdots,p_n\}$ は、ラベル l が表す文書内容が、$\{p_1,\cdots,p_n\}$ なる文書要素の集合で構成されることを意味する。

例 1

i) RDB: 搬入ＤＢを構造化述語で表す。

搬入DB#{伝票ID#6921, 取引先ID#597, 品名#ノート},
搬入DB#{伝票ID#6928, 取引先ID#595, 品名#ボールペン},
搬入DB#{伝票ID#6940, 取引先ID#597, 品名#ボールペン},
搬入DB#{伝票ID#6951, 取引先ID#587, 品名#鉛筆},

ii) XML: 取引先ＤＢを構造化述語で表す。

$$
\text{取引先}DB\# \left\{ \begin{array}{l} \text{取引先}\#\{ID\#\text{"597"}, \text{社名}\#\text{春日商店}\}, \\ \text{取引先}\#\{ID\#\text{"595"}, \text{社名}\#\text{山田文具}\}, \\ \text{取引先}\#\{ID\#\text{"587"}, \text{社名}\#\text{三省社}\} \end{array} \right\}
$$

次に、構造化述語上での推論を与え、関連づける。

定義 2 *(structured predicate logic)*

t を項、L をラベル、$\mathcal{C},\mathcal{D}$ を内容、P を構造述語とする。

項公理 $t \vdash t$

空要素公理 $\mathcal{C} \vdash \epsilon$

左ラベル $\dfrac{\mathcal{C} \vdash \mathcal{D}}{L\#\mathcal{C} \vdash \mathcal{D}}\ (l\#)$

右ラベル $\dfrac{\mathcal{C} \vdash \mathcal{D}}{L\#\mathcal{C} \vdash L\#\mathcal{D}}\ (r\#)$

左コンマ $\dfrac{\mathcal{C} \vdash \mathcal{D}}{P,\mathcal{C} \vdash \mathcal{D}}\ (l,)$

右コンマ $\dfrac{\mathcal{C} \vdash P \quad \mathcal{C} \vdash \mathcal{D}}{\mathcal{C} \vdash P,\mathcal{D}}\ (r,)$

左*contraction* $\dfrac{P,P,\mathcal{C} \vdash \mathcal{D}}{P,\mathcal{C} \vdash \mathcal{D}}\ (lC)$

右*contraction* $\dfrac{\mathcal{C} \vdash P,P,\mathcal{D}}{\mathcal{C} \vdash P,\mathcal{D}}\ (rC)$

コンマに関し、左および右*exchange*規則

□

直観的な意味づけ： $\mathcal{C} \vdash P$ は、文書 $\mathcal{C}$ が文書要素 P を支持することを表す。あるいは、文書 $\mathcal{C}$ が文書要素 P を含むことを表す。

背景にある基本的な思想は左側の文書に右側の文書要素が含まれている場合、左文書は右文書を支持すると見なすというものである。この直観的な意味づけに従って、構造化述語論理の推論規則、公理を説明する。まず、同じ文書は互いに支持する関係にある（項公理）。空要素文書は任意の文書に支持される（空要素公理）。もし文書内容 $\mathcal{C}$ が文書内容 $\mathcal{D}$ を支持するならば、$\mathcal{C}$ をタグでまとめたものも、$\mathcal{D}$ を支持する（左ラベル）。また、同じ仮定の元で、$\mathcal{D}$ をタグ L でまとめたものは、$\mathcal{C}$ が同じタグでまとめられたものによって支持される（右ラベル）。さらに、もし文書内容 $\mathcal{C}$ が 文書内容 $\mathcal{D}$ を支持するならば、左辺に文書 P を加えた文書内容も $\mathcal{D}$ を支持する（左コンマ）。また、文書内容 $\mathcal{C}$ が 文書 P を支持し、また文書内容 $\mathcal{D}$ を支持するならば文書内容 $\mathcal{C}$ は $\mathcal{D}$ に P を加えた文書内容も支持する（右コンマ）。文書内容に文書 P が重複して現れる場合、重複したものを除いても意味は変わらない (左および右 contraction)。また、文書内容中で文書要素の順番を変えても意味は変わらない (左および右 exchange)。

以下の命題は基本的な性質を表す。

$$\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{}{\text{取引先}ID\#597 \vdash \text{取引先}ID\#I}}{\text{品名}\#\text{ノート} \vdash \text{品名}\#X \quad \text{伝票}ID\#6921, \text{取引先}ID\#597 \vdash \text{取引先}ID\#I}\,(l,)}{\text{伝票}ID\#6921, \text{取引先}ID\#597, \text{品名}\#\text{ノート} \vdash \text{品名}\#X, \text{取引先}ID\#I}\,(r,)}{\text{搬入}DB\#\{\text{伝票}ID\#6921, \text{取引先}ID\#597, \text{品名}\#\text{ノート}\} \vdash \text{搬入}DB\#\{\text{品名}\#X, \text{取引先}ID\#I\}}\,(r\#)}{\text{搬入}DB\#\{\text{伝票}ID\#6921, \text{取引先}ID\#597, \text{品名}\#\text{ノート}\}, \cdots \vdash \text{搬入}DB\#\{\text{品名}\#X, \text{取引先}ID\#I\}}\,(l,)^n$$

図 4: 証明例

命題 1 $\mathcal{C}, \mathcal{D}, \mathcal{E}$ は内容、P は文書要素、L はラベルを表す。

i) 右コンマは以下と置き換えられる。

$$\frac{\mathcal{C} \vdash P \quad \mathcal{D} \vdash \mathcal{E}}{\mathcal{C}, \mathcal{D} \vdash P, \mathcal{E}}\,(r,)$$

ii) ラベル内の和集合とコンマの関係

$$L\#(\mathcal{C} \cup \mathcal{D}) \vdash L\#\mathcal{C}, L\#\mathcal{D}$$

逆は成り立たない。

命題 1 ii) は、一つのラベルでまとめられた内容は、ラベルをその内容の任意の部分集合に分配したものを支持するが、逆は支持しないことを意味する。すなわち左辺の方がこの論理では意味的に真に強いものを表し、左辺と右辺は等価でないことを意味する。

次の命題は、 5節応用例で使用した定理証明器のアルゴリズムを基礎づけている。

命題 2 $\Gamma \vdash L\#u$ *iff*

i) 構造化述語 $L\#u$ が構造化述語 $q \in \Gamma$ の部分構造として現れる *or*

ii) $L\#\Sigma$ が構造化述語 $q \in \Gamma$ の部分構造として現れ、かつ、 $u = \{e_i\}$ のすべての構造化述語に対し、$\Sigma \vdash e_i$ が成り立つ。

構造化述語論理の証明をクエリを想定した具体例で示す。

例 2 図 *4* は、文書内容を表す変数 X,I を使って、「搬入ＤＢ」の元で、「品名」および「取引先」ラベルを持つそれぞれの文書内容を証明しようとする例を表す。ここで $X =$ ノート, $I = 597$ とすることで証明が完成する。このように証明による問い合わせによって文書からさまざまな知識を簡潔にとりだすことが可能である。

例 3 構造化文書から証明によって情報を抽出する応用を考えた場合の特徴を示す。今、

$$P = a\#\{b\#b1, c\#\{d\#d1, e\#e1\}, c\#\{d\#d2, e\#e2\}\}$$

とする。

i) 構造内の位置に対する独立性が高い。ラベル b, e は木構造において異なる深さにある。

$P \vdash b\#X, e\#Y$ を満足する (X, Y) の解は、$(b1, e1), (b1, e2)$

ii) 目的に応じたラベルによる文脈指定可能。

- $P \vdash a\#\{d\#X, e\#Y\}$ を満足する (X, Y) の解は、$(d1, e1), (d1, e2), (d2, e1), (d2, e2)$ の *4* とおり。
- $P \vdash c\#\{d\#X, e\#Y\}$ を満足する (X, Y) の解は、$(d1, e1), (d2, e2)$ の *2* とおり。

4 線形論理による意味

ここでは、線形論理 [3] による意味づけを行なう。

構造述語と線形論理には以下のような単純な写像 e がある。

$$(\epsilon)^e = \top \qquad (t)^e = t$$
$$(L\#\mathcal{C})^e = (L \otimes (\mathcal{C})^e) \& (\mathcal{C})^e \qquad (P, \mathcal{C})^e = (P)^e \& (\mathcal{C})^e$$

ここで t は項、L はラベル、$\mathcal{C}$ は内容、P は構造化述語である。

定理 1 線形論理における論理的帰結を $\models$、また構造化述語を P、内容を $\mathcal{D}$ で表す。すると以下の関係が成り立つ。

$$\text{If } P \vdash \mathcal{D} \text{ then } (P)^e \models (\mathcal{D})^e$$

Proof. 証明木の長さに関する帰納法で証明。
i) If $t \vdash t$ then $t \models t$ かつ if $\mathcal{C} \vdash \epsilon$ then $\mathcal{C} \models \top$ は両辺とも公理から明らか。

ii) If $\mathcal{C} \vdash \mathcal{D}$ then $\mathcal{C}^e \models \mathcal{D}^e$ かつ if $\mathcal{C} \vdash P$ then $\mathcal{C}^e \models P^e$ が成り立つと仮定する。

左ラベル

$$\frac{\mathcal{C}^e \models \mathcal{D}^e}{L \otimes \mathcal{C}^e \,\&\, \mathcal{C}^e \models \mathcal{D}^e}\; l\&$$
$$(L \# \mathcal{C})^e \models \mathcal{D}^e$$

右ラベル

$$\frac{\dfrac{L \models L \quad \mathcal{C}^e \models \mathcal{D}^e}{L \otimes \mathcal{C}^e \models L \otimes \mathcal{D}^e}\; r\otimes}{L \otimes \mathcal{C}^e \,\&\, \mathcal{C}^e \models L \otimes \mathcal{D}^e}\; l\& \qquad \frac{\mathcal{C}^e \models \mathcal{D}^e}{L \otimes \mathcal{C}^e \,\&\, \mathcal{C}^e \models \mathcal{D}^e}\; l\&}{L \otimes \mathcal{C}^e \,\&\, \mathcal{C}^e \models L \otimes \mathcal{D}^e \,\&\, \mathcal{D}^e}\; r\&$$
$$(L \# \mathcal{C})^e \models (L \# \mathcal{D})^e$$

左コンマ

$$\frac{\mathcal{C}^e \models \mathcal{D}^e}{P \,\&\, \mathcal{C}^e \models \mathcal{D}^e}\; l\&$$
$$(P, \mathcal{C})^e \models \mathcal{D}^e$$

右コンマ

$$\frac{\mathcal{C}^e \models P \quad \mathcal{C}^e \models \mathcal{D}^e}{\mathcal{C}^e \models P \,\&\, \mathcal{D}^e}\; r\&$$
$$\mathcal{C}^e \models (P, \mathcal{D})^e$$

& に関し、contraction および exchange 規則が成り立つのは明らか。

定理 1 の逆方向の証明は将来に残している。

5 応用例

HTML ページの表データと XML 文書のデータ統合する実験を行なった。実験システム (図 5) は、prolog をベースに構造化述語の拡張などを行なったインタープリタ ('NetLog Interpreter')、および、情報源データと構造化述語間の変換を受け持つ情報ソース用のラッパ ('Wrapper') に大きく分かれる。本インタープリターに関してはここでは述べず、構造化述語利用の論理的な仕組みのみを説明する。

本例では、HTML で表された「搬入 DB」と XML 文書として表された「取引先 DB」の間でジョイン、およびプロジェクション操作を加え、結果を HTML 形式で出力するものを示す。

ここで $htmltabwp(db1.html)$ は $db1.html$ (図 1) 上の table 情報である「搬入 DB」を構造化述語に翻訳した結果、$xmlwp(db2.xml)$ は $db2.xml$ (図 2) の XML 要素「取引先 DB」を構造化述語に翻訳した結果を表すものとする。ジョイン操作は、変数 Y を共有する以下の証明を行なうことによって得ることができる。

$$\begin{array}{lcl} htmltabwp(db1.html) & \vdash & Table1\#\{\text{取引先}ID\#Y, \text{品名}\#Z\} \\ xmlwp(db2.xml) & \vdash & \text{取引先}\#\{ID\#Y, \text{社名}\#K\} \end{array}$$

プロジェクション操作 (ID 情報の除去) は上記結果を下記構造述語に移すことでなすことができる。

$$'join + projectiontest'\#\{\text{取引先社名}\#K, \text{名称}\#Z\}$$

この構造述語を HTML ラッパを用いて出力した結果が図 6である。

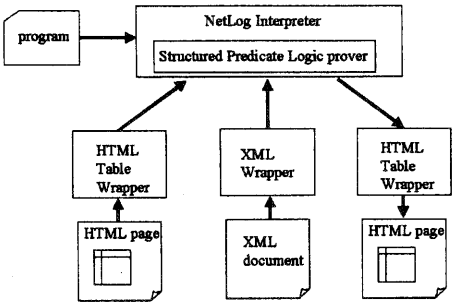

図 5: HTML, XML 統合

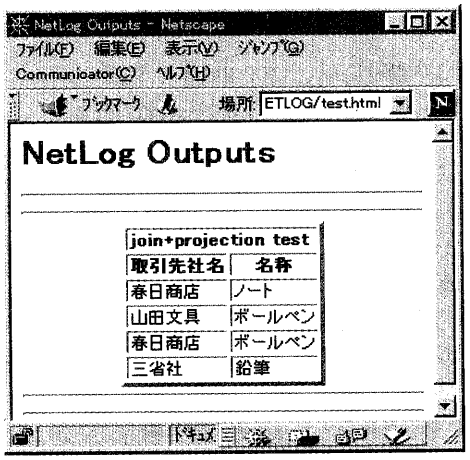

図 6: 出力 (HTML 形式)

References

[1] *ProDBI ODBC Interface for Quintus Prolog*, V4.0, Rob Lucas and Keylink Computers Ltd (1997).

[2] XML/SGML サロン: 標準 *XML* 完全解説, 技術評論社 (1998).

[3] Girard,J-Y: Linear Logic, *Theoretical Computer Science 50*, North-Holland, pp. 1-102, (1987).