3P−1

閉路を含む素性構造の単一化

李 航
日本電気(株) C&C情報研究所

落合尚良
日本電気技術情報システム開発(株)

1 はじめに

閉路を含まない素性構造の単一化による自然言語処理の手法がよく知られている[Shieber86]。しかし、実際の自然言語処理においては、より広い閉路を含む素性構造の単一化もしばしば必要である。例えば、タイプつき素性構造という拡張された素性構造[Aït-Kaci84]によって「私の買った本」という意味内容を表現する時に、閉路が現れてくる。従って、どのような特徴をもつ閉路を含む素性構造が単一化可能で、そのアルゴリズムがどうなるかというのは興味深い研究課題である。

本論文では、閉路を含む素性構造の単一化の可能性とその単一化アルゴリズムについて述べる。まず、有限オートマトンとして定義できる閉路を含む素性構造が必ず単一化可能であることを証明する。次に、そのような閉路を含む素性構造をタグリストというデータ構造として表現し、その単一化を行なうアルゴリズムを提案する。

2 単一化可能性

素性構造はラベルと値のペアの集合である。値はアトムあるいはさらに素性構造からなる。値にはタグが記されることがある。本研究では、ある値にあるタグが記された時、便宜的にその値のことをそのタグの値という。例えば、式1の素性構造においては、値[l2 a]にタグXが記されているため、[l2 a]をXの値という。

素性構造が有向グラフによって表現することができるので、以下では、有向グラフによって素性構造を表現することもある。例えば、式1、式2の素性構造をそれぞれ図1、図2の有向グラフによって表現する。

素性構造においては同名のタグの値が同一である。例えば、式1の素性構造においては、二つのタグXの値が同一で、共に[l2 a]である。タグの記し方によっては素性構造が閉路をもつことがある。例えば、式2の素性構造が閉路をもつ。

[加藤87]の指摘のとおり、閉路を含む素性構造を単一化した後、同名のタグの値が単一化された素性構造においても依然同一であることを如何にチェックするかが難点である。例えば、式1、2の二つの素性構造を単一化した結果、同名のタグの値が同一でないので、その単一化が失敗する。その例からもわかるように、タグチェックの仕方によっては処理が停止しない恐れがある。

$$\begin{bmatrix} l1 & & X \\ l2 & \begin{bmatrix} l2 & a \end{bmatrix} & X \end{bmatrix} \quad (1)$$

$$\begin{bmatrix} l1 & \begin{bmatrix} l2 & Y \end{bmatrix} & Y \end{bmatrix} \quad (2)$$

従って、どのような特徴をもつ素性構造が単一化可能であるかを研究する必要がある。以下では、閉路を含む素性構造を有限オートマトンとして定義し、そのような素性構造が単一化可能であることを証明する。

定義 1 有限オートマトン$A = (Q, \Sigma, \Gamma, \delta, q_0, F, \lambda, R)$を定義する。それは素性構造であるとする。$Q$は有限状態の集合で、$\Sigma$はラベルの可算集合で、$\Gamma$はアトムの可算集合で、$\delta: Q \times \Sigma \to Q$は状態遷移関数(部分関数)で、$q_0 \in Q$は初期状態で、$F \subseteq Q$は最終状態の集合で、$\lambda: F \to \Gamma$は出力関数である。$R$は以下の条件を満足する$Q$における反射的対称的推移的関係である。

$$pRq \Leftrightarrow \delta(p, l) = \delta(q, l), \forall l \in \Sigma^* \quad (3)$$

関係Rは有限オートマトンにおいて同名のタグの値が同一であることを規定する。本研究では同名のタグの値が同一であるかどうかをチェックすることをタグチェックという。

定理 1 有限オートマトン理論によれば、ある正則集合を受理する有限オートマトンを、状態数の最も少ない一つの有限オートマトンにする(最小化する)ことができる[Hopcroft84]。以上の有限オートマトンにおけるタグチェックは普通の有限オートマトンの最小化に相当する。

証明: $\delta(q_0, l_p) = p$、$\delta(q_0, l_q) = q$、さらにpRqであれば、$\delta(p, l) = \delta(q, l), \forall l \in \Sigma^*$である。つまり、$l_p$と$l_q$が右不変関係である。よって、状態$p$と$q$を一つの状態にまとめることができる。つまり、以上の有限オートマトンにおけるタグチェックは普通の有限オートマトンの最小化に相当する。

系 1 タグチェックの計算量は$\leq O(kn^2)$である。但し、nは有限オートマトンの状態数で、kはラベル数である。

定義 2 $A = (Q_A, \Sigma_A, \Gamma_A, \delta_A, q_{0A}, F_A, \lambda_A, R_A)$
および$B = (Q_B, \Sigma_B, \Gamma_B, \delta_B, q_{0B}, F_B, \lambda_B, R_B)$
が有限オートマトンであるとする。AがBを包摂する($A \preceq B$)とは、以下の準同形写像$h: Q_A \to Q_B$が成り立つことである。

$$\begin{aligned} &h(q_{0A}) = q_{0B} \\ &\forall q \in Q_A, h(\delta_A(q, l)) = \delta_B(h(q), l) \\ &\forall q \in F_A, \lambda(q) = \lambda_B(h(q)) \\ &\forall p, q \in Q_A, pR_Aq \Rightarrow h(p)R_Bh(q) \end{aligned} \quad (4)$$

二つの有限オートマトン(素性構造)の単一化は以上に示す関係の最小上界(lub)を求める操作として定義する。本研究では、二つの有限オートマトンのq_0からみて対応できる状態をまとめることを有限オートマトンの併合という。

系 2 二つの有限オートマトン(素性構造)の単一化は、二つの有限オートマトンを併合し、併合された有限オートマトンを最小化することによって実現できる。

3 単一化アルゴリズム

以上に示すように、素性構造を有限オートマトンとして定義し、単一化することができる。しかし、実際、素性構造を

Unification of Cyclic Feature Structures
Hang LI
C&C Information Technology Research Laboratories, NEC.
Takayoshi OCHIAI
NEC Scientific Information System Development, Ltd.

有限オートマトンとして表現するのは、処理の効率がよくない。本研究では、素性構造をタグリストというデータ構造として表現し、単一化を行なうことを提案する。タグリストとは、タグとその値のペア[1]のリストのことである。タグリストにおいて0という仮想的なタグが素性構造全体に記されるとする。例えば、図1、図2の素性構造をタグリストとして表現すると、それぞれ図3、図4のようになる。素性構造をタグリストとして表現することは素性構造をタグを中心に分割することに相当する。

タグリストとして表現される素性構造の単一化は併合とタグチェックという二つの操作によって実現する。併合、或はタグチェックが失敗したら、単一化が失敗する。

タグリストの併合は以下のアルゴリズムによって行う。

ステップ1. 二つのタグリストをAppendする。タグリストの中に同名のタグをもつタグ値ペアがあれば、それらをタグリストから取り出し、それらに対してステップ2を実行する。なければ、タグリストの併合が終了する。

ステップ2. 取り出したタグ値ペアを一つに併合する。併合されたタグ値ペアを元のタグリストに戻す。タグ値ペアを併合することは、すなわちタグ値ペアの値に対して、従来の意味の単一化を行なうことである。具体的には、それぞれの対応するラベルの下の値が同一であれば、併合が成功する。さもなければ、併合が失敗する。併合の際、それぞれの値の中のタグが重なることがある。重なったタグの中の一つのタグ名を採用し、タグリストにある重なったタグの名を全部そのタグ名に統一する。

ステップ3. ステップ1へ。

重なったタグの名の統一は、重なったタグを大域的に保持し、以降の処理で常にそれを参照していくという実現しかたも考えられる。

タグリストにおけるタグチェックは以下のアルゴリズムによって行う。

ステップ1. タグリストにあるタグ値ペアに対して左から右へ順番にステップ2を実行していく。実行し終ったら、タグチェックが終了する。

ステップ2. 注目するタグ値ペアΓの値の中のタグを、順番に以下のようにチェックしていく。

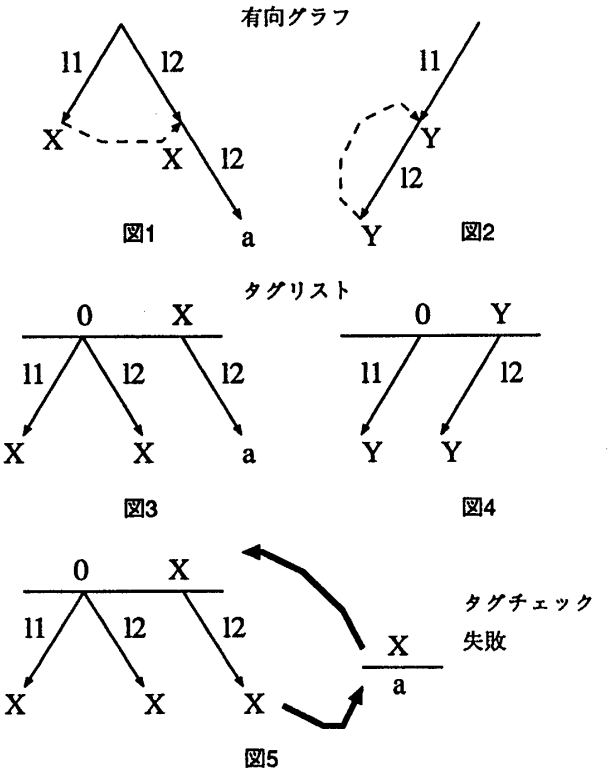
ステップ2-1. あるタグAの値が「空」である時、タグAをとばす。

ステップ2-2. あるタグAの値がαである時、αを取り出し、タグAの値を「空」にする。と同時に、タグAとαにより新しいタグ値ペアΣを作成する。

ステップ2-2-1. タグ値ペアΣのタグと同名のタグをもつタグ値ペアがタグリストにない時、タグ値ペアΣをタグリストの一番右に追加する。

ステップ2-2-2. タグ値ペアΣのタグと同名のタグをもつタグ値ペアがタグリストにある時、二つのタグ値ペアを併合する。併合の結果によっては、注目するタグ値ペアΓより左側にあるチェック済みのタグ値ペアに影響を与えることがある。影響を受けたタグ値ペアをタグ値ペアΓより右側に移動する。タグ値ペアΓに対してもう一度ステップ2を実行する。

例えば、図3と図4のタグリストを併合すれば、図5のタグリストが得られる。次に図5のタグリストに対してタグチェックを行なう。タグリストにおいてX : [l2 a X]がタグ値ペアで、さらにその値の中にタグXの値がアトムaであるので、新しいタグ値ペアX : aが得られる。その二つのタグ値ペアを併合することができず、単一化が失敗する。



本研究では、以上で示す単一化アルゴリズムを計算機の上で実現し、そのアルゴリズムによる閉路を含む素性構造の単一化が可能であることを確かめた。

4 おわりに

本論文では、まず、有限オートマトンとして定義できる閉路を含む素性構造が必ず単一化可能であることを証明した。次に、そのような閉路を含む素性構造をタグリストというデータ構造で表現し、単一化を行なうアルゴリズムについて提案した。

参考文献

[Aït-Kaci84] H. Aït-Kaci. *A Lattice Theoretic Approach to Computation Based on a Calculus of Partially Ordered Type Structures, PhD Thesis, Univ. Pennsylvania, 1984.*

[Hopcroft84] J. ホップクロフト, J. ウルマン 著, 野崎昭弘 等訳. オートマトン 言語理論 計算論 *1*, サイエンス社, *1984.*

[加藤87] 加藤進, 小暮潔. 素性構造の単一化手法の効率. 情報処理学会, 自然言語処理研究会 *64-9, 1987.*

[Shieber86] S. M. Shieber. *An Introduction to Unification-Based Approaches to Grammar, CSLI Lecture Notes 4, 1986.*

[1] 以下ではタグとその値のペアのことを「タグ値ペア」という。