3ＡＨ－7

相関にもとづく論理プログラムの自動合成 一般のHorn規則の合成方法*

大木 敏幸† 加藤 進也‡ 中村 克彦¶
東京電機大学大学院理工学研究科§ 東京電機大学理工学部‖

1 はじめに

論理プログラミングの計算は融合(resolution)と呼ばれる演繹推論規則にもとづいた述語論理の定理証明プロセスである. これに対し, 最近, 逆融合(inverse resolution)などの帰納推論規則をもちいて入出力例から論理プログラム自動合成するための研究が進められている [1, 2].

われわれは, 相関と呼ばれるプロセスによって多数の入出力例を解析し, その結果から論理プログラムを自動合成する方式と, それにもとづくシステムSYNAPS (Synthesis by Analyzing Positive Samples)の研究を続けている [3]. 今回の報告では, これまでに報告した相関を改良することによって, より一般的な規則を含む論理プログラムを合成が可能であることを示す.

2 相関(Correlation)

この報告では、次のような記号を用いる.

定数：	$a, b, c, \cdots$
変数：	$X, Y \cdots$
定数の並び：	$\overline{A}, \overline{B}, \cdots$
変数の並び：	$\overline{X}, \overline{Y}, \cdots$

任意のアトム $p(t_1, \cdots, t_n)$ に対して, 項 $p(t'_1, \cdots, t'_n)$ を**部分アトム**(subatom)と呼ぶ. ここで, 各 t'_i は t_i の部分項である.

*Automatic Synthesis of Logic Programs by Correlation - Synthesis Method for General Horn Rule
†Toshiyuki OOKI
‡Shinya KATO
§Graduate School of Science and Engineering , Tokyo Denki University
¶Katsuhiko NAKAMURA
‖The Factory of Science and Engineering , Tokyo Denki University

[例] $p([a,b],[c])$ の部分アトムは以下の6個である.
$p([a,b],[c]), p([a,b],[]), p([b],[c]), p([b],[]), p([],[c]), p([],[])$.

最小汎化LGG(Least General Generalization)は単一化(unification)の逆演算であり, 多くの上向き方式の帰納推論の基礎となっている [1]. 任意のアトムP, Qに対してLGGの結果は(R, δ_1, δ_2)である. ここで, $R\delta_1 = P$ かつ $R\delta_2 = Q$ であり, かつRはこの条件を満足する項の中でもっとも具体化されたものである.

相関とは, あるアトムの内部にほかのアトムの構造が含まれていることをLGGを用いて検出するアルゴリズムであり, 1対の入出力例をあらわす単位節からこれに含まれる再帰的構造を解析するために使われる.

相関は, 任意のアトム $p(\overline{S})$,$p(\overline{T})$ に対して適用され, 以下に定義される6組

$$(p(\overline{D_1}), p(\overline{D_2}), p(\overline{U}), \delta_1, \delta_2, SF)$$

の集合を結果とする.

$p(\overline{T})$ の部分項$p(\overline{T'})$ と $p(\overline{S})$ の部分項$p(\overline{S'})$ の各対に対して, 次のように相関の結果が与えられる.

1. **front difference** (FD) $p(\overline{D_1}), p(\overline{D_2})$
   $p(\overline{D_1})$ は, $p(\overline{T'}) = p(t_1, \cdots, t_n)$ としたとき, $p(f_1(t_1), \cdots, f_n(t_n)) = p(T)$ となるような項 $p(\overline{D_1}) = p(f_1(X_1), \cdots, f_n(X_n))$ と定義される. $p(\overline{D_2})$ は, $p(\overline{S'})$ に対して同様に定義される.

2. **common part** (CP) $p(\overline{U})$ および **back difference** (BD) δ_1, δ_2は, $p(\overline{T'})$ および $p(\overline{S'})$ に対するLGGの結果 $(p(\overline{U}), \delta_1, \delta_2)$ によって与えられる.

3. **simirality factor** $SF = f - v$. ここで, fは $p(\overline{U})$ に含まれる変数と定数の出現の個数, v は$p(\overline{U})$ の変数と定数の種類の個数である.

上に述べた自己相関を拡張して, 異なる述語の例に対する相互相関を定義できる. 相互相関は, 与えられた述語の例と論理プログラムによって back ground 知識として定義済みの述語との相互関係を導く.

3 節の合成

相関の結果によって合成されるHorn節には, 単位節, および次の3種類の規則がある.

$$
\begin{array}{ll}
\text{PP型} & p(\overline{D}) \leftarrow p(\overline{A}). \\
\text{PQ型} & p(\overline{D}) \leftarrow q(\overline{A}). \\
\text{PPQ型} & p(\overline{D}) \leftarrow p(\overline{A}), q(\overline{B}).
\end{array}
$$

ここでは, PP型規則, PQ型規則の生成法のみについて述べる.

PP型規則の合成　相関の結果の解析でBDが空のとき, すなわち $(p(\overline{D_1}), p(\overline{D_2}), p(\overline{U}), \phi, \phi, SF)$ の形のとき, 次の規則を生成する.

$$p(f_1(X_1), \cdots, f_n(X_n)) \leftarrow p(f_1'(X_1), \cdots, f_n'(X_n))$$

ここで$p(\overline{D})$ はFDの一般項, 各x_iは$p(\overline{D})$ 中の対応する引数の含む変数である.

[回文の文法のDCGの例]　2つの入力例 $p([b,a,b|Z_1], Z_1)$ と $p([a,b,a,b,a|Z_2], Z_2)$ に対して, 次の相関の結果が得られる.

$$
\begin{array}{ll}
FD: & p(\overline{D_1}) = p([a|X_1], X_2), \\
 & p(\overline{D_2}) = p(X_1, X_2), \\
CP: & p(\overline{U}) = p([b,a,b|Y_1], Y_2), \\
BD: & \delta_1 : Y_1 = [a|Z_2], Y_2 = Z_2, \\
 & \delta_2 : Y_1 = Y_2 = Z_1.
\end{array}
$$

この結果から、次の規則が生成される.

$$p([A|X], Y) \leftarrow p(X, [A|Y]).$$

PQ型規則の合成　相関の結果がFDが空でかつBDが空でないとき, すなわち $(p(\overline{X_1}), p(\overline{X_2}), p(\overline{U}), \delta_1, \delta_2, SF)$ の形のとき, 新しい述語g/nを含む次のような規則および2つの単位節 (g/nの例) が生成される.

$$
\begin{array}{l}
p(f_1(X_1), \cdots, f_n(X_n)) \leftarrow q(X_1, \cdots, X_n). \\
q(f_1(X_1), \cdots, f_n(X_n))\delta_1. \\
q(f_1(X_1), \cdots, f_n(X_n))\delta_2.
\end{array}
$$

ここで, $p(\overline{U}) = p(f_1(X_1), \cdots, f_n(X_n))$ である.

[正規言語のDCGの例]　2つの入力例 $p([a,b,a])$ と $p([a,b,a,b,a])$ に対する相関の結果は次のようになる.

$$
\begin{array}{l}
p(\overline{D_1}) = p(\overline{D_2}) = p(X), \\
p(\overline{U}) = p([a,b,a|X]), \\
\delta_1 = [], \ \delta_2 = [b,a].
\end{array}
$$

この結果から, 次のような規則と単位節の例が生成される.

$$
\begin{array}{l}
p([a,b,a|X]) \leftarrow q(X). \\
q([]). \\
q([b,a]).
\end{array}
$$

4 おわりに

今回の報告では, これまでの相関のアルゴリズムを改良し, PQ型規則の合成オペレータをつけくわえたことにより, より一般的な規則の合成が可能となることを示した. 今後は, ここで述べた自己相関プロセスに, 相互相関プロセスをつけくわえ, さらに, 多数の相関の結果を解析する機能を持たせたシステムを作成していきたいと考えている.

参考文献

[1] F.Bergadano and D.Gunetti : Inductive Logic Programming from Machine Learning to Software Engineering, The MIT Press, 1996.

[2] S.Muggleton : Inductive Logic Programming, Academic Press, 1992.

[3] 中村 克彦 : 帰納推論による論理プログラムの自動合成に関する調査研究報告書, 財団法人機械システム振興協会, 1993.