LF_004

知識の関係構造を用いた新しい関数の生成
Generating Functions from Relational Structure

金盛 克俊 † 延澤 志保 ‡ 太原 育夫 ‡
Katsutoshi Kanamori Shiho Nobesawa Ikuo Tahara

1 はじめに

新たな知識や概念の獲得は人工知能における主要なテーマの一つである．近年，大量のデータから有意な知識を発掘するデータマイニングに関する研究が盛んに行われ，こうした知識発見の方法を基礎付け体系化する構想も検討されている [1].

知識発見や発見学習という言葉はデータマイニングとともに用いられることが多くなっているが，知識発見とデータマイニングは必ずしも同意語ではない [2]．知識発見はデータマイニングに限ったものではなく，より広い概念である.

本稿では知識に現れない新しい概念の生成，発見について考える．一般に，述語論理式で表現された知識において，概念とは述語を指すことが多いが，関数や個体は返り値の存在性と唯一性が保障された概念と考えることができる．与えられた知識やデータから述語を形成するシステム [3][4][5] や，特化した表現の概念を生成するシステムはいくつか提案されている [6][7][8] が，関数や個体を生成する手法についてはほとんど考慮されていない．存在記号を冠頭にもつことが許されない知識においては，関数，個体を生成することで述語の生成では表現できない存在性と一意性を持った新しい概念を生成することができる.

本稿では，知識の上位概念である関係構造を用いて，与えられた知識に現れない新しい関数，個体を生成する手法を提案する．個体は 0 引数の関数として捉えると，関数生成の流れは図 1 のようになる．まず，既存の知識またはその一部を一般化することによってそこに内在する関係構造を抽出する．関係構造は述語や関数の自由変数をもつ論理式として定義される．そして，抽出した関係構造に現れるある関数変数を新しい具体的な関数と見做し，それ以外の変数には知識に現れる具体的な述語や関数を代入することによって関係構造を具体化し，新たな関数の現れる節が生成されることになる.

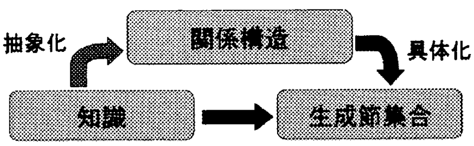

図 1: 関数生成の流れ

本稿では，特に関係構造と関数変数が与えられているときに関数を生成する部分を議論の対象とする．さらに関数を生成することにより生じる矛盾を解消するために対象領域を拡張することによって，より創造的な知識処理が実現できることを示す.

2 概念と関係構造

2.1 概念

関数は，2つのものが全く同じものであることを意味する述語=（イコール）と存在記号∃を用いれば，述語を用いて書き換えることができ，さらに個体は 0 引数の関数と考えることができる．関数は，例えばその引数とそれに対する返り値を引数とした述語を用いて表現することができる．例えば n 変数関数 f に対して $F(x_1, x_2, \cdots, x_n, y)$ の真理値が，$f(x_1, x_2, \cdots, x_n) = y$ ならば真，そうでなければ偽となるような $n+1$ 変数述語 F を用いると，f の現れる論理式は，関数 f が現れず述語 F のみが現れる論理的に同値な論理式に変換することができる．個体を引数なしの関数と考えることで個体も同様に変換することができる．以上の理由から，本稿では関数や個体も概念と考え，概念生成の対象とする.

2.2 知識と関係構造

知識として，第一階閉述語論理式のうち節形式（頭部がすべて全称記号で母式が連言標準形である冠頭標準形）で表現されたものを考え，その母式から得られる節の集合を知識と呼ぶことにする．関係構造は，自由な述語変数，自由な関数変数，自由な個体変数のうち少なくとも一つの現れる節からなる集合である．これらの変数をまとめて概念変数と呼ぶことにする．一般に，述語変数を含み，関数変数および個体変数を含まない論理式はスキーマと呼ばれる [9]．しかしここでは，関数変数と個体変数を含むことを許したスキーマを関係構造として定義する.

2.3 単純代入

関係構造を具体化する操作に必要な，自由変数を定数に置き換える単純代入を定義する．ここで定義する単純代入は変数を項に置き換える一般の代入と違い，単に記号の置き換えを行うだけである．述語変数と述語定数の対，関数変数と関数定数の対，個体変数と個体定数の対を要素としてもつ集合として単純代入を定義する．n 引数の自由述語変数全体の集合を P_{vn}，n 引数の述語定数全体の集合を P_{cn} とおく．同様に n 引数の自由関数変数の集合を F_{vn}，n 引数の関数定数の集合を F_{cn}，自由個体変数の集合を U_v，個体定数の集合を U_c としたとき，

$$\begin{aligned}\Theta \subset\ & P_{v1} \times P_{c1} \cup \cdots \cup P_{vi} \times P_{ci} \cup \cdots \cup \\ & F_{v1} \times F_{c1} \cup \cdots \cup F_{vj} \times F_{cj} \cup \cdots \cup U_v \times U_c\end{aligned}$$

であるような Θ のうち，同じ変数，定数が 2 つ以上現れないものを単純代入という．Θ の要素を一般的な記法に倣って変数 v と定数 c に対して v/c と書く．特に関係構造 RS に対して，RS に現れる定数の現れない Θ を重複

†東京理科大学大学院理工学研究科情報科学専攻
‡東京理科大学理工学部

のない単純代入という．以降では関係構造に対して単純代入 Θ が与えられているとき，特に断りがないかぎり Θ は重複のない単純代入であるとする．

節 c と単純代入 Θ に対して，c に Θ を適用した節 $c\Theta$ は c に現れる変数を Θ で指定された定数に置き換えた節として定義される．節の集合 S に対して $S\Theta$ は S の全ての要素に Θ を適用した集合である．

さて，関係構造を用いて新しい知識を生成したとき矛盾が生じないためには，少なくとも関係構造自体が無矛盾でなければならない．そこで関係構造における矛盾を以下のように定義する．まず，節集合 Σ から節 c が導出操作によって導かれるとき $\Sigma \vdash_d c$ と表記することとし，関係構造 RS から節 c が導出されるということを RS および c に現れる自由変数を RS に現れない別々の定数と見做したとき $RS \vdash_d c$ であると定義する．関係構造においても誤解を生じないかぎり1階の述語論理式と同じ記号を用いる．関係構造 RS が矛盾しているとは，RS に現れる自由変数を RS に現れない別々の定数とみなしたときに RS が矛盾していることをいう．RS が矛盾しているとき，RS に現れる自由変数が全て現れるどんな単純代入 Θ においても $RS\Theta$ から空節が導かれる．

3 新しい関数の現れる生成節

以降では個体を0引数の関数と捉え，誤解のないかぎり関数には個体も含むものとする．新しい関数は，新しい関数が現れる新しい節が生成されることにより得られる．そのような節の集合を生成節と呼び，その集合を生成節集合と呼ぶことにする．知識 Σ，関係構造 RS，及び RS に現れる関数変数 Y が与えられたとき，生成節集合 S_{new} は以下のように定義される．

定義 1 Σ に現れる述語定数の集合を P_Σ，関数定数（個体定数を含む）の集合を F_Σ とする。RS に現れる述語変数の集合を P_{RS}，Y 以外の自由な関数変数（全称記号で束縛された個体変数は含まず，自由な個体変数を含む）の集合を F_{RS} とする．$RS^* = \{c | RS \vdash_d c\}$ としたとき，RS^* の要素の中の Y の現れる節について，Y で始まる項を全て冠頭の全称記号で束縛された別々の新しい変数に置き換えた（同じ項は同じ変数に置き換える）節の集合を S_Y，RS^* の要素の中で Y の現れない節の集合を $S_{\bar{Y}}$ とする．P_{RS} の全ての要素に P_Σ の要素を，F_{RS} の全ての要素に F_Σ の要素をそれぞれ重複なく対応させていく。このような単純代入は有限個であるので，それらを $\Theta_1, \Theta_2, \cdots, \Theta_n$ とする．このとき，新規性条件

$$\forall c \in S_Y\Theta_i : c \notin Th(\Sigma)$$
$$\forall c \in S_{\bar{Y}}\Theta_i : c \in Th(\Sigma)$$

を満たす Θ_i によって得られる生成節の集合を

$$S_{new_i} = \{c_Y(\Theta_i \cup \{Y/new_i\}) | c_Y \in RS, c_Y \text{は} Y \text{が現れる節}\}$$

と定義し，これらの総和集合を S_{new} とする．

このように定義された S_{new} の要素である生成節は，新規性条件の $\forall c \in S_Y\Theta_i : c \notin Th(\Sigma)$ によって，新たな関数を含み，既存の知識の定理とならない新しい節であることが保障される．すなわち，以下のような定理が得られる．

定理 1 知識 Σ，関係構造 RS，関数変数 Y が与えられたときの生成節の集合を S_{new} とすると，

$$S_{new} \cap Th(\Sigma) = \phi$$

である．

例 1 自然数と0に関する次のような知識 Σ を考える．

$$\Sigma = \left\{ \begin{array}{l} \neg > (y,0) \vee > (+(x,y),x) \\ > (x,0) \vee = (x,0) \\ \neg = (1,0) \end{array} \right\}$$

これに，述語変数 X_0 に関して対称的な関数変数 f と g の現れる関係構造 RS を考える．

$$RS = \left\{ \begin{array}{l} \neg X_0(f(x,y),x) \vee \neg X_0(g(x,y),x) \\ \neg X_1(g(a,b),a) \end{array} \right\}$$

このとき Σ，RS，g について関数を生成する．

$$S_g = \left\{ \begin{array}{l} \neg X_0(f(x,y),x) \vee \neg X_0(z,x) \\ \neg X_1(x,a) \end{array} \right\}, S_{\bar{g}} = \phi$$

であるから，

$$\Theta_1 = \{X_0/>, X_1/=, f/+, a/0, b/1\}$$

は新規性条件を満たすので，以下の S_{new_1} が生成される．

$$S_{new_1} = \left\{ \begin{array}{l} \neg > (+(x,y),x) \vee \neg > (new_1(x,y),x) \\ \neg = (new_1(0,1),0) \end{array} \right\}$$

加算 $+$ と対照的な関数である減算のような新しい関数 new_1 が生成されたことがわかる．

4 関数生成による対象領域の拡張

3章で定義した生成節をそのまま知識に加えると矛盾することがある．生成された新しい関数の返り値の存在を必ず仮定することになるため，それがもとの知識の対象領域に存在しないものであるときに矛盾するのである．

今，新たな1変数の述語 inD を用いて Σ の各節 c_j に対して

$$\neg inD(x_{j1}) \vee \cdots \vee \neg inD(x_{jl_j}) \vee c_j$$

を考える．ただし，$x_{j1}, \cdots, x_{jl_j}$ は c_j に現れる全ての束縛変数である．全ての Σ の要素におけるそのような節の集合と，S_{new_i} の和集合を Σ' とする．すなわち

$$\Sigma' = \left\{ \begin{array}{l} \neg inD(x_{11}) \vee \cdots \vee \neg inD(x_{1l_1}) \vee c_1 \\ \vdots \\ \neg inD(x_{n1}) \vee \cdots \vee \neg inD(x_{nl_n}) \vee c_n \end{array} \right\} \cup S_{new_i}$$

である．もとの知識 Σ の対象領域を D，Σ' の対象領域を D' とし，$inD(x)$ の解釈を $x \in D$ ならば真，$x \notin D$ ならば偽と考えれば明らかに $D \subseteq D'$ であり，無矛盾性に関する以下の定理が得られる．

定理 2

$$\Sigma' = \left\{ \begin{array}{l} \neg inD(x_{11}) \vee \cdots \vee \neg inD(x_{1l_1}) \vee c_1 \\ \vdots \\ \neg inD(x_{n1}) \vee \cdots \vee \neg inD(x_{nl_n}) \vee c_n \end{array} \right\} \cup S_{new_i}$$

としたとき，Σ' は無矛盾である．

Σ' は新しい関数 new_i の返り値の存在を全て仮定した知識になっていると考えられる．さらに，必要ならば Σ に現れる個体定数 $a_1, \cdots, a_m$ に対して

$$inD(a_1), \cdots, inD(a_m)$$

を Σ' に加えてもよい．この場合も定理 2 と同じ証明で Σ' が無矛盾であることがいえる．このような述語 inD は必ず必要なわけではないが，定理より無矛盾であることが保障できる．

以下の例は自然数と 0 に関する知識において，加算と対照的な関数である減算のような関数を生成することにより，対象領域を拡張した例である．

例 2 先の例 1 において，生成された S_{new_1} は Σ と矛盾する．すなわち $\Sigma \cup S_{new_1}$ からは空節が導出される．そこで述語 inD を用いて以下のような Σ' を新たな知識とすると，Σ' は new_1 の返り値の存在を仮定し Σ の対象領域を拡張した知識となっていることがわかる．

$$\Sigma' = \left\{ \begin{array}{l} \neg inD(x) \vee \neg inD(y) \vee \neg > (y, 0) \vee > (+(x, y), x) \\ \neg inD(x) \vee > (x, 0) \vee = (x, 0) \\ \neg = (1, 0) \\ \neg > (+(x, y), x) \vee \neg > (new_1(x, y), x) \\ \neg = (new_1(0, 1), 0) \\ inD(0) \\ inD(1) \end{array} \right\}$$

この Σ' は定理より無矛盾である．Σ' からは，例えば $\neg inD(new_1(0,1))$ などが導出され，new_1 の生成によって拡張された知識における新たな定理が得られることがわかる．

5 生成節を求めるアルゴリズム

生成節集合を定義に従いそのまま計算すると一般に計算は停止しない．RS から導出操作によって RS^* を求める際，RS^* は有限集合とは限らないため停止することが保障されないからである．さらに，新規性条件を調べる計算について，$c \in Th(\Sigma)$ かどうかを調べるには $\Sigma^* = \{c | \Sigma \vdash_d c\}$ としたとき

$$\exists c_0 \in \Sigma^* : c_0 \succeq c$$

であるかを調べればよいが，この操作も全ての Σ^* の要素について c を包摂するかどうかを調べることになるので，Σ^* が一般に無限集合になることから有限時間で求めることができない．

ところで，ある節 c が節 c_0 に包摂されるかを調べる操作は，$c_0\theta \subseteq c$ となるような θ が存在するかどうかを調べればよい．c の変数に対して c_0 に現れる項を置き換える代入は有限通りしかないので，それぞれについて $c_0\theta \subseteq c$ を調べる操作は有限回で終わる．従って，RS^*，Σ^* が有限集合であれば有限時間の計算で生成節を求めることができる．

しかし RS^*，Σ^* は一般に有限ではなく，それを調べる計算も有限時間で終わるとは限らないので，実際には RS と Σ に導出操作を有限回行って RS^* と Σ^* に近づけて近似するのが自然であると考えられる．そこで

$$\begin{array}{ll} \Sigma \subseteq \Sigma' \subseteq \Sigma^* & (\forall c \in \Sigma' : \Sigma \vdash_d c) \\ RS \subseteq RS' \subseteq RS^* & (\forall c \in RS' : RS \vdash_d c) \end{array}$$

であるような Σ' と RS' を用いて，S_{new} を近似した S'_{new} を求めるアルゴリズムを以下に示す．Σ，RS に対して導出する導出節の数の上限をそれぞれ N_Σ，N_{RS} とする．

アルゴリズム

1. $\Sigma' = \Sigma$，$RS' = RS$，$S'_{new} = \phi$ とする．

2. Σ' の 2 つの要素から導出節を求め Σ' に加える．この操作を繰り返し，それ以上導出節が得られないか，得られた導出節の個数が N_Σ に達したら 3 へ．

3. RS' の 2 つの要素から導出節を求め RS' に加える．この操作を繰り返し，それ以上導出節が得られないか，得られた導出節の個数が N_{RS} に達したら 4 へ．

4. RS' の要素の中の Y の現れる節について，Y で始まる項を全て冠頭の全称記号で束縛された別々の新しい変数に置き換えた節の集合を S'_Y，RS' の要素の中で Y の含まれない節の集合を $S'_{\bar{Y}}$ とする．RS に現れる Y 以外の変数を Σ に現れる定数に置き換える全ての単純代入を考える．そのような単純代入は有限個であるので，$\Theta_1 \cdots \Theta_n$ とする．Θ_i に対して，

$$\begin{array}{lcl} \forall c \in S'_{\bar{Y}}\Theta_i \forall c_0 \in \Sigma' & : & c_0 \not\succeq c \\ \forall c \in S'_{\bar{Y}}\Theta_i \exists c_0 \in \Sigma' & : & c_0 \succeq c \end{array}$$

であるならば，

$S'_{new_i} =$
$\{c_y(\Theta_i \cup \{Y/new'_i\}) | c_y \in RS, c_y$は Y が現れる節 $\}$

とし，S'_{new} を $S'_{new} \cup S_{new_i}$ とする．これを全ての Θ_i に対して行う．

このようにして得られた S'_{new} は，$RS' = RS^*$ かつ $\Sigma' = \Sigma^*$ のとき，明らかに $S'_{new} = S_{new}$ である．そして S'_{new} は有限時間で求めることができる．しかし，一般的には新規性条件を満たさない節が生成される可能性がある．このことに関連して，Σ' と RS' の性質と S'_{new} との関係について以下の定理が得られる．

定理 3 $\Sigma' = \Sigma^*$ のとき，すなわち Σ^* が有限集合ならば，本来生成されるべき節の集合 S_{new} に対して

$$S_{new} \subseteq S'_{new}$$

である．

この定理は，Σ^* が求められれば，新規性条件を満たさない節が生成されることがあるが，本来生成されるべき生成節は全て求められることを示している．

定理 4 $S'_Y \Theta_k = \Sigma'$ であるような単純代入 Θ_k が存在するとき $\Sigma \cup S'_{new_k}$ は無矛盾である．

定理4は，条件を満たす単純代入については生成節をそのまま知識に加えても矛盾を起こすことはないことを示している．逆に言うとこのような場合には新たな述語 inD を用いた対象領域の拡張ができない．この定理は，生成節を対象領域の拡張なしに元の知識に加えるのか，関数生成により対象領域を拡張するのかを定めるための基準となり得る重要な定理である．

6 まとめと課題

関数や個体を知識における概念として位置づけ，関係構造を用いて新しい関数を生成する手法を提案した．これにより，従来の概念生成では実現できなかった柔軟な概念獲得が可能となる．それに伴い，新たな領域の存在を仮定し，矛盾なく知識の対象領域を拡張する方法を提案した．

実際のシステムでどのような関数を生成すればよいかは目的により異なるが，生成節は知識，関係構造，関数変数から一意に定まるため，知識に対してどのように関係構造を抽出しその変数に着目するかという問題が今後の課題となる．そのような知識，関係構造，関数変数を選ぶ指針が定まり，与えられた知識から関数を自動的に生成することができれば，知識を自動的に拡張していく創造的な知識処理を行うシステムの構築が期待される．

本稿は未知のものの存在を仮定した場合どうなるかを考えるという，人間が自然に行う知的活動の1つであり従来の概念学習や概念生成では実現できなかった，柔軟で全く新しい知識創出の方法を提案するものである．

参考文献

[1] 有川節夫, 佐藤雅彦, 佐藤泰介, 丸岡章, 宮野悟, 金田康正： 発見科学の構想と展開, 人工知能学会誌, Vol.15, No.4, pp.595–607 (2000).

[2] 元田浩：知識発見・データマイニング, 人工知能学辞典, pp.648–658, 人工知能学会 (2005).

[3] Muggleton, S. and Buntime, R.: Machine Invention of first-order Predicates by Inverting Resolution, in Proc. of the 5th Intl. Workshop on Machine Learning, pp.339–352, ANN Arbor, MI (1988).

[4] Kijsirikul, B., Numao, M. and Shimura, M.: Discription-Based Constructive Induction of Logic Programs, AAAI-92, pp. 44–49, AAAI Press/MIT Press(1992).

[5] 金盛克俊, 延澤志保, 太原育夫：関係構造を用いた述語の生成, 電子情報通信学会 2004 年総合大会講演論文集, 情報・システム 1, 104 (2004).

[6] 嶋田晋 ： ヒューリスティクスにより発見を実現するシステム, 人工知能学会誌, Vol.2, No.1, pp.52–61 (1986).

[7] Davis, R. and Lenat, D .B.: Knowledge-Based System in Artificial Intelligence, McGraw-Hill, NewYork (1982)

[8] Lenat, D.B.: The Role of Heuristics in Learning by Discovery : Three case studies , in, R.S. Michalski, Learning, Tioga Pub. Co., Palo Alto (1983)

[9] 山田敬三, 平田耕一, 原尾政輝：スキーママッチングとその計算量, 電子情報通信学会論文誌 D-I Vol.182-D-I, No.11 pp.1307–1316 (1999).