平成18年度 修士論文概要		
帰納学習のための述語のメタ的性質と その自動生成に関する研究	和田・犬塚研究室 No. 18417507	石田 浩之 Hiroyuki Ishida

1 はじめに

帰納論理プログラミング(Inductive Logic Progamming:ILP) [1] は述語論理表現を用い，帰納推論を行う枠組みである．ILP は分類学習やデータマイニングに広く応用可能である．

ILP での分類学習では，目標とする概念 (目標概念) の事例に対し，これを説明する理論（＝論理式の集合）を得る．目標概念に当てはまる事例（正事例）と概念に当てはまらない事例（負事例），さらに理論を構成する他の概念の定義（背景知識）が与えられ，事例を説明する理論 (仮説) を得る．

仮説の候補は事例集合によって評価され，簡単なものから複雑なものへ順に構成されるトップダウン方式がよく採用される．このとき評価する必要の無い仮説の候補の数を抑える仮説生成制御が重要である．従来，入出力モードや型情報などを用いた仮説生成制御法が知られる．先行研究として，石田らは述語のメタ的性質を用いた一般的な仮説生成制御法を提案している．本研究では，この制御法で用いられるメタ的性質を形式化した上で，メタ的性質で仮説を制御する方法の原理を明らかにする．さらに与えられた外延的な知識ベースから述語のメタ的性質を自動的に生成する方法を提案する．

2 ILP 分類学習とメタ的性質

ILP における分類学習の例を示す．目標概念が $\mathrm{add}(A,B,C)$ であり，自然数 A,B,C に対して $A+B=C$ となることを示すとしよう．この場合 $\langle 1,1,2\rangle$, $\langle 2,1,3\rangle$ などが正事例，$\langle 1,1,1\rangle$, $\langle 2,1,1\rangle$ などが負事例である．背景知識 $\mathcal{B}$ には $\mathrm{add}(A,B,C)$ を説明する材料として，$\mathrm{inc}(A,B)$（$A+1=B$), $\mathrm{dec}(A,B)$ $(A-1=B)$, $\mathrm{one}(A)$ $(A=1)$ が与えられているとする．この場合，次式は $\mathrm{add}(A,B,C)$ の事例を説明している．ILP 分類学習器はこうした理論を出力する．

$$\begin{aligned}\mathrm{add}(A,B,C) &\leftarrow \mathrm{one}(B)\wedge\mathrm{inc}(A,C)\\ \mathrm{add}(A,B,C) &\leftarrow \mathrm{inc}(A,D)\\ &\qquad\wedge\mathrm{dec}(B,E)\wedge\mathrm{add}(D,E,C)\end{aligned}$$

理論の探索は，リテラルと論理式の追加で行われる．論理式にリテラルを加えると条件がきつくなり，負事例を排除できる．満たすべき正事例が満たされないときは論理式を新たに追加する．追加すべきリテラルの選択は，事例によって評価される前に，述語のメタ的性質を利用して評価できる．たとえば，論理式に $\mathrm{inc}(A,B)$ が含まれているとき，$\mathrm{dec}(A,B)$ とは両立しない，$\mathrm{dec}(B,A)$ が成り立つ，値 B は値 A より大きいといったことが直ちにわかる．さらに，ここで用いた値の大小関係には推移律が成り立つ．したがって，論理式に $\mathrm{inc}(A,B)$ が含まれているとき，$\mathrm{dec}(A,B)$ も $\mathrm{dec}(B,A)$ も追加できない．また，さらに $\mathrm{inc}(B,C)$ が含まれていれば大小関係の推移性から $\mathrm{dec}(C,A)$ とも両立しない．

3 メタ的性質による仮説生成制御

我々はこれまでに述語のメタ的性質を用いた仮説生成制御法を提案した．この制御法で述語のメタ的性質を扱うために，リテラルに関連して 3 つの条件を与える．前節で inc が値の大小に関してメタ的性質（この場合，order とする）を持つように，直接背景知識に現れない述語を用いて記述する必要がある．

基礎的前提　メタ的性質を記述する述語において常に成り立つ論理式
例：order をメタ的性質として扱うにはその推移性（$\mathrm{order}(X,Z)\leftarrow\mathrm{order}(X,Y)\wedge\mathrm{order}(Y,Z)$）が基礎的前提として必要である．

付加的前提　対象のリテラルを論理式に加えたときメタ的性質として成立する事実.
例：inc(A,B) に対して，order(A,B) は成り立つので付加条件に含まれる．

禁止条件　対象のリテラルを論理式に加えるためには成立してはならない事実.
例：inc(A,B) を加えるには，dec(A,B) や order(B,A) はあってはならず，禁止条件に含まれる．

これらの条件等を用いて仮説探索時，節にリテラルを加えるべきかどうかを判断する．節 $C=g\leftarrow l_1\wedge\cdots\wedge l_n$ にリテラル l' を加える場合，式 1 を確認する．これが成り立つ場合，C に l' を加えない．成り立たない場合は C に l' を加え，その後の事例を使った評価に移る．

$$\left(\bigcup_{i=1}^{n} l_i\text{の付加的前提}\right)\cup\text{基礎的前提} \vdash l'\text{の禁止条件の要素} \qquad (1)$$

4 メタ的性質の形式化

事例やそれを説明する背景知識など，ここで扱う論理式に用いる述語記号，関数記号，定数記号にそれの数学的意味を割り当てることを**解釈**という．知識ベースの中で各記号は特定の意味を持っており，これにしたがって解釈されなければならない．これを**意図的解釈**という．解釈 I のもとで論理式 P が真となることを $I\models P$ と表記する．

メタ的知識を扱う場合，知識ベースに現れない記号の中でメタ的性質を記述するのに必要な記号も含めて考えなければならない．ここではそれらの記号も含めて意図的解釈を持っていると考える．このときメタ的性質は次のように定義できる．

定義 1 (メタ的性質) 与えられた知識ベースとそれに関連するメタ的性質のための述語の意図的解釈を $\mathcal{I}$ とする．このとき $\mathcal{I}\models\mathcal{M}$ を満たし，関数記号，定数記号を含まない論理式集合をメタ的性質と呼ぶ．

提案する仮説生成制御法　変数の組を$\boldsymbol{x}$, $\boldsymbol{x}_1$ 等で表記する．リテラル l や論理式 F がもつすべての変数が$\boldsymbol{x}$であるとき，$l(\boldsymbol{x})$, $F(\boldsymbol{x})$ と書く．

また，変数の組$\boldsymbol{x}$について，各変数毎に新しい定数記号を用意した定数の組を $n(\boldsymbol{x})$ と書くことにする．変数の組$\boldsymbol{x}$を持つ $l(\boldsymbol{x})$ に対し，$l(n(\boldsymbol{x}))$ は変数を持たないリテラルである．このとき次の定理を示した．

定理 1 (メタ的性質の拡張) $\mathcal{M}$ を与えられた知識ベースのメタ的性質，$l_1(\boldsymbol{x}_1), \cdots, l_1(\boldsymbol{x}_n)$，および，$l'(\boldsymbol{x}')$ を定数記号，関数記号を含まないリテラルとするとき，

$$\mathcal{M} \cup \{l_1(n(\boldsymbol{x}_1)), \cdots, l_n(n(\boldsymbol{x}_n))\} \vdash l'(n(\boldsymbol{x}'))$$

が成り立つならば $l_1(\boldsymbol{x}_1) \wedge \cdots \wedge l_n(\boldsymbol{x}_n) \to l'(\boldsymbol{x}')$ はメタ的性質である．

変数を定数記号に置換えた式での導出を使ってメタ的性質を調べることができる．定理 1 より系 1，2 が成り立つ．

系 1 (冗長) $F(\boldsymbol{x})$ を連言形の論理式，その部分式を $F'(\boldsymbol{x}')$，n を $\boldsymbol{x}$ の要素数，$l(\boldsymbol{x}'')$ をあるリテラルとする．ただし，$\boldsymbol{x}'$ は $\boldsymbol{x}$ の一部，$\boldsymbol{x}''$ は $\boldsymbol{x}'$ の一部である．このとき，$F'(\boldsymbol{x}') \to l(\boldsymbol{x}'')$ がメタ的性質ならば，
$\{\boldsymbol{x} \in |\mathcal{I}|^n \mid \mathcal{I} \models F(\boldsymbol{x})\} = \{\boldsymbol{x} \in |\mathcal{I}|^n \mid \mathcal{I} \models F(\boldsymbol{x}) \wedge l(\boldsymbol{x}'')\}$ である．

系 2 (矛盾) $F(\boldsymbol{x})$ を連言形の論理式，その部分式を $F'(\boldsymbol{x}')$，n を $\boldsymbol{x}$ の要素数，$l(\boldsymbol{x}'')$ をあるリテラルとする．ただし，$\boldsymbol{x}'$ は $\boldsymbol{x}$ の一部，$\boldsymbol{x}'''$ は $\boldsymbol{x}$ と $\boldsymbol{x}''$ を集めた変数の組である．このとき，$F'(\boldsymbol{x}') \wedge l(\boldsymbol{x}'') \to \Box$ がメタ的性質ならば，
$\{\boldsymbol{x}''' \in |\mathcal{I}|^n \mid \mathcal{I} \models F(\boldsymbol{x}) \wedge l(\boldsymbol{x}'')\} = \emptyset$ である．

系 1，2 から制御法は次のようになる．

節 $C = g \leftarrow l_1 \wedge \cdots \wedge l_n$ にリテラル l' を加える場合，下記の二つの条件の内どちらかが成り立つ場合，C に l' を加えず，そうでなければ C に l' を加える．

1. $\mathcal{M} \cup \{l_1(n(\boldsymbol{x}_1)), \cdots, l_n(n(\boldsymbol{x}_n))\} \vdash l'(n(\boldsymbol{x}'))$. ここで $\boldsymbol{x}'$ は $\boldsymbol{x}_1, \cdots, \boldsymbol{x}_n$ を集めた変数組の一部.
2. $\mathcal{M} \cup \{l_1(n(\boldsymbol{x}_1)), \cdots, l_n(n(\boldsymbol{x}_n)), l'(n(\boldsymbol{x}'))\} \vdash \Box$

5 メタ的性質の自動生成

外延データーベースを利用したメタ的性質の自動生成法を提案する．二つの手続きからなる．main_gene の入力は仮説の生成およびメタ的性質に用いる全述語の集合 $\mathcal{P}$ と $\mathcal{P}$ の述語全ての外延データーベース $\mathcal{DB}$ である．このとき，$\mathcal{DB}$ を $\mathcal{P}$ の意図的解釈とみなすことができる．出力は $\mathcal{P}$ の述語を使ったメタ的性質 $\mathcal{M}$ である．sub_gene の入力は連言形式の論理式 F と main_gene と同様の述語集合 $\mathcal{P}$ および外延データーベース $\mathcal{DB}$，出力は $\mathcal{P}$ の述語のみを使い，各要素に F が使われているメタ的性質 $\mathcal{M}$ である．

$\mathcal{M} \leftarrow \emptyset$
for each $p/n \in \mathcal{P}$
　$\mathcal{M} \leftarrow \mathcal{M} \cup sub\_gene(p(X_1, \cdots, X_n), \mathcal{P}, \mathcal{DB})$

図 1:main_gene のアルゴリズム

$\mathcal{M} \leftarrow \emptyset$
F と少なくとも変数を一つ共有するリテラルの集合 $\mathcal{L}$ を生成
for each $l \in \mathcal{L}$
　if F の変数の集合 $\supseteq l$ の変数の集合 then
　　if $\{\boldsymbol{x} \mid \mathcal{DB} \models F(\boldsymbol{x})\} = \{\boldsymbol{x} \mid \mathcal{DB} \models F \wedge l(\boldsymbol{x})\}$ then
　　　$\mathcal{M} \leftarrow \mathcal{M} \cup \{l \leftarrow F\}$
　　if $\{\boldsymbol{x} \mid \mathcal{DB} \models F \wedge l(\boldsymbol{x})\} = \emptyset$ then
　　　$\mathcal{M} \leftarrow \mathcal{M} \cup \{\Box \leftarrow F \wedge l\}$
　else
　　$\mathcal{M} \leftarrow \mathcal{M} \cup sub\_gene(F \wedge l, \mathcal{P}, \mathcal{DB})$

図 2:sub_gene のアルゴリズム

6 実験

5 節で提案したアルゴリズムを実装し，メタ的性質の生成試験を行い，メタ的性質の要素数および実行時間を比較した．メタ的性質の各要素におけるリテラルの最大数が 2 個と 3 個それぞれの場合のメタ的性質の要素数および実行時間は表 1 に示す．

表 1：メタ的性質の要素数と実行時間

最大リテラル数 [個]	2	3
メタ的性質の要素数 [個]	208	23860
実行時間 [sec]	0.266	38.656

この結果より，最大リテラル数が 1 増加すると要素数および実行時間が急激に増加することが分かる．原因究明のために出力内容を検証した結果，非冗長性および網羅性が満たされていないことが判明した．

7 まとめと今後の課題

本研究では，三つのことを行った．一つ目がメタ的性質の定義，二つ目が仮説生成制御法の再提案および正当性の証明，三つ目がメタ的性質の自動生成法の提案および動作実験である．今後，自動生成法を改良し，仮説生成制御の動作実験を行う必要がある．

参考文献

[1] J.R.Quinlan，R.M.Cameron-Jones :“ Introduction of Logic Programs:FOIL and Related Systems ”，New Generation Computing，vol 13，pp.287-312，OHMSHA，1995

[2] J.R.Quinlan :“ Learning Logical Definitions from Relations ”，Machine Learning，5，pp.239-266，1990

発表論文

- 石田 浩之，中野 智文，犬塚 信博 :” 述語のメタ的性質を利用した帰納学習の仮説制御”，第二回 情報科学ワークショップ，2006
- 石田 浩之，中野 智文，犬塚 信博 :” 帰納学習のための述語のメタ的性質とその自動生成に関する研究”，第 151 回 SIG-ICS，2008（発表予定）