

1N-5 負例から求めた msc による Progol の仮説探索の効率化

中谷 唯[†] 近山 隆^{††}

[†]東京大学大学院工学系研究科 ^{††}東京大学大学院新領域創成科学研究科

1 概要

帰納論理プログラミング (ILP) は、1 階述語論理上の帰納推論システムである。ILP では、入力として与えられた正例、負例、背景知識の条件を満たす仮説を、一般・特殊の半順序関係で結ばれた仮説候補のなす概念束 (仮説空間) 内を探索することにより求める。代表的 ILP システム Progol では、1 つの正例から最弱仮説 (msc) を求め、これをボトムとする部分束内を効率的に探索している。本発表では、Progol において従来正例だけを用いてきた msc のほかに、負例から求めた msc を合わせて用いることにより、仮説空間が縮小可能であることを示す。

2 ILP システム Progol の仮説探索

帰納論理プログラミング (Inductive Logic Programming, ILP) は、一階述語論理の部分系である論理プログラムを知識表現とする、帰納推論の一種である。

帰納推論は演繹推論と対をなしており、「具体的事例が与えられるとき、その事例を説明する一般法則を導き出すプロセス」である。

以下ではとくに断らない限り、 X 、 Y 、 \dots を変数として、家族関係を例に ILP を説明する。 X が Y の「おじさん」であるという関係を導くことを考え、図 1 に示すデータ (正例、負例、背景知識) を ILP システムの一つ Progol に与えて実行すると次のようなことを意味する答えが返ってくる。

$$X \text{ is uncle_of } Y \leftarrow X \text{ is brother_of } Z, \\ Z \text{ is parent_of } Y.$$

ILP システムでは、一般・特殊の半順序関係で候補仮説が結ばれた仮説空間の中を探索する。Progol では、探索に先立ち正例の一つを選んでそれを伴意

Target relation : uncle_of

Positive examples :

カツオ is uncle_of タラオ.
海平 is uncle_of サザエ.
海平 is uncle_of カツオ.
波平 is uncle_of ノリスケ.

Negative examples :

タラオ is uncle_of カツオ.
波平 is uncle_of カツオ.
ワカメ is uncle_of タラオ.

Background knowledge :

海平 is brother_of 波平.
波平 is father_of サザエ.
波平 is father_of カツオ.
波平 is father_of ワカメ.
サザエ is mother_of タラオ.
カツオ is brother_of サザエ.
ワカメ is sister_of サザエ.
波平 is brother_of 波平の妹.
波平の妹 is mother_of ノリスケ.

$X \text{ is parent_of } Y. \leftarrow X \text{ is father_of } Y.$

$X \text{ is parent_of } Y. \leftarrow X \text{ is mother_of } Y.$

$X \text{ has same_parent as } Y. \leftarrow X \text{ is brother_of } Y.$

$X \text{ has same_parent as } Y. \leftarrow X \text{ is sister_of } Y.$

図 1: Description of family relation

する仮説の中で最も特殊な仮説 (最弱仮説 : msc) を求め、探索すべき仮説空間を縮小している [1]。また Progol では A^* アルゴリズムに基づく探索によって仮説を求める。

3 負例の利用による仮説探索の効率化の可能性

Progol では msc 生成の際に正例を用いているが、一方の負例は仮説の評価に用いるだけで探索空間の縮小には利用されてこなかった。本研究では正例と似たような形をした負例を利用して、より効率的に仮説を探索することができないか考える。

背景知識 B とある例 e により求められる msc を $msc(B, e)$ と書くと、

$$e_1^+ = \text{uncle_of}(\text{カツオ}, \text{タラオ}).$$

について $msc(B, e_1^+)$ は

$$\text{uncle_of}(X, Y)$$
$$\therefore \text{same_parent}(X, Z), \text{brother_of}(X, Z),$$
$$\text{parent_of}(Z, Y), \text{mother_of}(Z, Y).$$

Improving the Search Algorithm of Progol with msc's Built on Negative Examples

Tadashi Nakaya and Takashi Chikayama

The University of Tokyo

7-3-1 Hongo, Bunkyo, Tokyo 113-8656 Japan

となる。従来 msc を求める際には例として正例のみを用いていたが、与えられた 3 番目の負例

$e_3^- = \text{uncle_of}(\text{ワカメ}, \text{タラオ}).$

について $\text{msc}(B, e_3^-)$ を求めると、

$\text{uncle_of}(X, Y)$

$:- \text{same_parent}(X, Z), \text{sister_of}(X, Z),$

$\text{parent_of}(Z, Y), \text{mother_of}(Z, Y).$

となる。この $\text{msc}(B, e_3^-)$ を $\text{msc}(B, e_1^+)$ と比較するとその違いは述語 1 つのみであり (bother_of が sister_of になっている)、これは e_1^+ と e_3^- の違いを反映しているものと考えられる。すると、 $\text{msc}(B, e_1^+)$ にのみ存在する $\text{brother_of}(X, Z)$ は、求める仮説の本体部の一部となる可能性が高いのではないかと推定することができる。そこで仮説空間の探索の際にこの $\text{brother_of}(X, Z)$ に関連する仮説候補から探索を始めることで、早いうちに高い評価値が得られ、枝を切るすることができるので、結果として計算時間を短縮することができるものと思われる。特にこの場合は

$\text{uncle_of}(X, Y) :- \text{brother_of}(X, Z).$

より特殊な仮説のみを探索すればよい (図 2)。

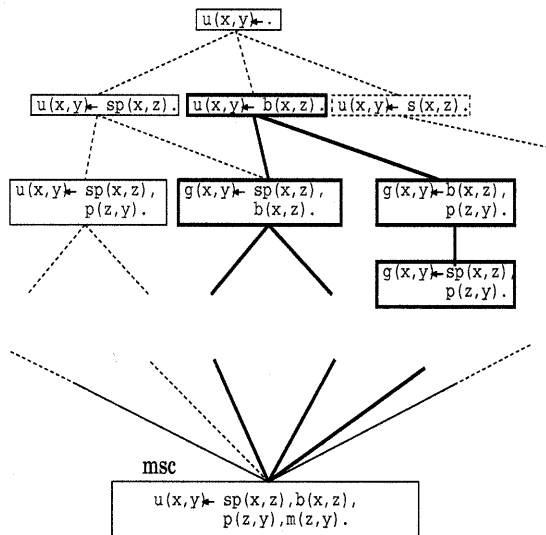


図 2: Reduced hypothesis space

4 効率的な仮説探索の手順

以上を踏まえて、効率的な仮説探索の手順としておおよそ以下のようなものが考えられる。

1. 与えられたすべての正例、負例について msc を求める。
2. 正例の msc と負例の msc 1 つずつを選び、それらに用いられている述語の種類などからその類似度を計算し、最も類似度の高い組を求める。
3. 2. で求めた組について、正例の msc に含まれ負例の msc に含まれないリテラルを求める。
4. 3. で求めたリテラルが優先されるように、 A^* アルゴリズムに基づく仮説探索を行なう。

各々の具体策については、今後の課題である。

5 まとめと今後の方針

本稿では、ILP の概要とその主なシステムの一つである Progol について述べたあと、負例を利用して仮説空間を縮小することにより、計算量の縮小を実現できる可能性があることを示し、そのための手順の外枠を述べた。

今後はまず先に述べた効率的な仮説探索の具体的な手順の構築を目指し、多くの問題で本手法の有効性を検証する。

また与えられた正例に対して、効率的な仮説探索を行なうことができるような負例の選出法についても検討する予定である。

さらに、本手法では同時に正負 2 つの例から msc を求めてその差異に注目しているが、正例 2 つから求めた msc の共通部分に注目して、同様の効率化は図れないだろうか、GOLEM[2] で用いられる最小汎化との関連性と共に考えたい。

参考文献

- [1] Muggleton, S. : “Inverse Entailment and Progol,” New Generation Computing, Special issue on Inductive Logic Programming 13 (3 - 4), 1995
- [2] Muggleton, S. and Feng, C. : “Efficient Induction of Logic Programs,” Proceedings of the First International Workshop on Algorithmic Learning Theory, Tokyo, 1990, pp. 368 - 381