

帰納論理プログラミングを用いた化学反応からのルール抽出 Rule Extraction from Chemical Reaction Using Inductive Logic Programming

力規晃† 越村三幸† 橋本司† 山下全広‡ 藤田博† 長谷川隆三†
Noriaki Chikara Miyuki Koshimura Osamu Hashimoto Masahiro Yamashita Hiroshi Fujita
Ryuzo Hasegawa

1. はじめに

化学反応を利用して、反応物 (reactant) から生成物 (product) を効率よく生成するには、反応物の濃度や流量、また、温度など、反応を取り巻く様々な因子・条件 (以後、パラメータ) を適切に設定する必要がある。適切なパラメータは、実験を繰り返し行うことによって求めるのが普通であるが、起こりうる全ての場合を実験で網羅することは、現実的には不可能である。少ない実験回数で、最適なパラメータを求めることが工学的には重要である。

各パラメータが、反応物と生成物の関係に及ぼす影響について、定性的に把握できれば、最適なパラメータを求める際の助けになると考えられる。本研究では、実際の化学実験データから、帰納論理プログラミングを用いてパラメータの影響の定性的な性質の抽出を試みた。

2. 化学反応のパラメータと生成物

本研究で対象とする化学反応において考慮するパラメータは4つあり、 $P_{in}^1, \dots, P_{in}^4$ と標記する。生成物は2つあり、その収量を P_{out}^1, P_{out}^2 と標記する。解析に先立って、これらはいずれも0以上1以下の実数値に正規化される。

目標とするのは、投入する資源量を最小化しつつ、得られる資源量を最大することである。これは、正規化後のデータでは、 $P_{in}^1, \dots, P_{in}^4$ をいづれもできるだけ小さくし、 P_{out}^1, P_{out}^2 の両方をできるだけ大きくする、と言い換えられる。

実験データ数は96件である。つまり、パラメータを変えて96通りの実験を行った。そして得られた $\langle P_{in}^1, \dots, P_{in}^4, P_{out}^1, P_{out}^2 \rangle$ の実数値の組96個を使いルール抽出を行った。

3. 帰納論理プログラミングの適用

本研究では帰納論理プログラミング (Inductive Logic Programming, 以下 ILP)[1]を用いてパラメータ $P_{in}^1, \dots, P_{in}^4$ と収量 P_{out}^1, P_{out}^2 の間に成り立つルール抽出を行う。

3.1 帰納論理プログラミング

ILPとは一階述語論理に基づいた機械学習の手法である。ILPの一般的な枠組みは、正例 E^+ と負例 E^- 、背景知識 B が節集合で与えられ、

$$\begin{cases} B \models E^+ \\ B \cup E^- \not\models \square \end{cases} \quad (1)$$

であるとき、

$$\begin{cases} H \cup B \models E^+ \\ H \cup B \cup E^- \not\models \square \end{cases} \quad (2)$$

を満たす仮説 H を見つけることである。本研究ではILPシ

†九州大学 Kyushu University

‡東洋紡績株式会社 TOYOBO CO., LTD.

ステムとしてProlog上で動作するAleph[2]を用いる。

3.2 データの量子化

実数値のデータを離散的に扱うため量子化を行う。元の領域を10段階のレベルに分けることを目安に区分し、名前を付ける。パラメータ P_{in}^X の値がレベル n であることを $\text{in}X_n$ (X は1~4の整数, n は0~9の整数)と表現する。ただし、実験に用いたパラメータは元々離散的な値であり、 P_{in}^1 は2通り、 P_{in}^2 は3通り、 P_{in}^3 は4通り、 P_{in}^4 は6通りの値を取る。そして、それぞれ表1(a)~(d)のように名前を付ける。

収量についても同様に P_{out}^X の値がレベル n であることを $\text{out}X_n$ (X は1~2の整数, n は0~9の整数)と表現する。実験では、 P_{out}^1 は0.1510以上0.6500以下、 P_{out}^2 は0.188167以上0.639667以下の範囲に収まった。それぞれ、表1(e)(f)のように10等分した。

表1 パラメータ及び収量の量子化

(a) P_{in}^1 の量子化

元の値	0.833333	0.888889
量子化後の表現	in1_0	in1_9

(b) P_{in}^2 の量子化

元の値	0.125	0.25	0.5
量子化後の表現	in2_0	in2_3	in2_9

(c) P_{in}^3 の量子化

元の値	0.1	0.25	0.5	0.7
量子化後の表現	in3_0	in3_2	in3_6	in3_9

(d) P_{in}^4 の量子化

元の値	0.2	0.3	0.4	0.6	0.8	1
量子化後の表現	in4_0	in4_1	in4_2	in4_4	in4_7	in4_9

(e) P_{out}^1 の量子化

元の値の範囲	[0.1510, 0.2009]	[0.2009, 0.2508)	[0.2508, 0.3007)	[0.3007, 0.3506)	[0.3506, 0.4005)
量子化後の表現	out1_0	out1_1	out1_2	out1_3	out1_4
元の値の範囲	[0.4005, 0.4504)	[0.4504, 0.5003)	[0.5003, 0.5502)	[0.5502, 0.6001)	[0.6001, 0.6500]
量子化後の表現	out1_5	out1_6	out1_7	out1_8	out1_9

(f) P_{out}^2 の量子化

元の値の範囲	[0.188167, 0.233317]	[0.233317, 0.278467)	[0.278467, 0.323617)	[0.323617, 0.368767)	[0.368767, 0.413917)
量子化後の表現	out2_0	out2_1	out2_2	out2_3	out2_4
元の値の範囲	[0.413917, 0.459067)	[0.459067, 0.504217)	[0.504217, 0.549367)	[0.549367, 0.594517)	[0.594517, 0.639667)
量子化後の表現	out2_5	out2_6	out2_7	out2_8	out2_9

量子化後の表現については帰納論理プログラミングの枠組みでのアトムとして取り扱う。

3.3 正負例と背景知識

本研究が対象とする実験データ数は 96 件である。収量をできるだけ多くするのが目標であるので、望ましい実験データは、 P_{out}^1 が out1_9 であり、かつ、 P_{out}^2 が out2_9 であると考えられる。このような事例は 11 件あり、これらを正例とした。そして他の 85 事例を負例とした。

背景知識は 96 件分の実験データを述語表現したものを与える。本研究で用いる述語表現を表 2 に示す。

表 2 本研究で用いる知識表現

知識表現	意味
input1(E, In)	事例 E の P_{in}^1 は In である。
input2(E, In)	事例 E の P_{in}^2 は In である。
input3(E, In)	事例 E の P_{in}^3 は In である。
input4(E, In)	事例 E の P_{in}^4 は In である。
output1(E, Out)	事例 E の P_{out}^1 は Out である。
output2(E, Out)	事例 E の P_{out}^2 は Out である。

3.4 追加の背景知識

パラメータがある値以上、あるいは、ある値以下であることを扱う表 3 のような知識表現を導入する。例えば P_{in}^2 がレベル 1 以上ならば、

input2LessOrEqual(Ex,in2_1) :- input2(Ex,in2_0).

input2LessOrEqual(Ex,in2_1) :- input2(Ex,in2_1).

のようなルールを背景知識に追加する。このように各々の条件に関して、必要なルールを背景知識に追加する。

表 3 以上、以下を扱う知識表現

知識表現	意味
input1GreaterOrEqual(E,In)	事例 E の P_{in}^1 は In 以上
input1LessOrEqual(E,In)	事例 E の P_{in}^1 は In 以下
input2GreaterOrEqual(E,In)	事例 E の P_{in}^2 は In 以上
input2LessOrEqual(E,In)	事例 E の P_{in}^2 は In 以下
input3GreaterOrEqual(E,In)	事例 E の P_{in}^3 は In 以上
input3LessOrEqual(E,In)	事例 E の P_{in}^3 は In 以下
input4GreaterOrEqual(E,In)	事例 E の P_{in}^4 は In 以上
input4LessOrEqual(E,In)	事例 E の P_{in}^4 は In 以下

3.5 実験データの解析

本研究では Prolog システムは SWI-Prolog 5.6.63[3]を用い、その上で ILP システム Aleph[2] Version5 を実行した。

結果として得られたルールを図 1 に示す。要した計算時間は数秒程度であった。ここで、Pos cover 値と Neg cover 値はそれぞれ、このルールを満たす正例数と負例数を表す。

Rule 1 は P_{in}^2 のレベルが 3 かつ P_{in}^4 のレベルが 7 のとき望ましい収量が得られていることを意味している。そして、Rule 1 を満たす正例は 6 個ある。また、Rule2 は P_{in}^3 のレベルが 9、 P_{in}^4 のレベルが 9、かつ P_{in}^2 のレベルが 3 以下であ

るときに望ましい収量が得られることを示している。そして、4 つの正例が Rule 2 を満たしている。Rule3 は、Rule1 と Rule2 のいずれも満たさない正例が 1 つあることを示している。

```
[Rule 1] [Pos cover = 6 Neg cover = 0]
example(A) :-
  input2(A, in2_3), input4(A, in4_7).
[Rule 2] [Pos cover = 4 Neg cover = 0]
example(A) :-
  input3(A, in3_9), input4(A, in4_9),
  input2LessOrEqual(A, in2_3).
[Rule 3] [Pos cover = 1 Neg cover = 0]
example(ex071).
```

図 1 得られたルール

4. 考察

$P_{in}^1, \dots, P_{in}^4$ をいづれもできるだけ小さく、つまりこれらのレベルをできるだけ低くし、 P_{out}^1, P_{out}^2 の両方をできるだけ大きくするという目標にそって、Rule1 と Rule2 を比較してみよう。Rule1 と Rule2 を満たす事例は、いずれも「望ましい収量」を得ている。この収量を得るために Rule2 では P_{in}^3 と P_{in}^4 のレベルを 9 つまり最大にする必要があるが、Rule1 では P_{in}^4 はレベル 7 でよく、 P_{in}^3 は無条件となっている。 P_{in}^2 については、Rule1 ではレベル 3、Rule2 ではレベル 3 以下と、Rule 2 の方が若干低いレベルとなっている。これらを総合的に考えると Rule1 の条件がより優れていると考えられる。

このように実験データからは、 P_{in}^2 のレベルが 3 で、 P_{in}^4 のレベルが 7 の時、望ましい収量が得られる、ことが導かれる。

5. おわりに

パラメータを様々に変えて化学反応の実験を行い、得られた実験データに ILP を適用し、望ましい収量を得るパラメータの条件をルールとして抽出した。そして、実際に実験室で実現し得る適切なルールを得ることができた。

本研究では、パラメータと収量間の定性的な関係を導いたが、実際には、定量的な関係も知りたい。そのためには、回帰分析といった統計的手法が有用であると思われ、今後これらの手法の適用も試みていきたい。

謝辞

本研究の一部は科研費 (21300054) の助成をうけたものである。

参考文献

- [1] 古川 康一, 尾崎 知伸, 植野 研, “帰納論理プログラミング”, 共立出版 (2001)
- [2] A.Srinivasan, “The Aleph Manual”, <http://www.comlab.ox.ac.uk/oucl/research/areas/machlearn/Aleph/> (1999)
- [3] “SWI-Prolog’s home”, <http://www.swi-prolog.org/>