

# Generalization System, Regular pattern language, Minimal Language and $k$ -multiple minimal generalization

January 14, 2015

## 参考

- Polynomial time inference of extended regular pattern languages (Shinohara)
  - [http://link.springer.com/chapter/10.1007/3-540-11980-9\\_19](http://link.springer.com/chapter/10.1007/3-540-11980-9_19)
  - Regular pattern の minimal common genelization (mcg) を多項式時間で求める
- Finding Minimal Genelization for Unions of Pattern Languages and ... (Arimura+)
  - [http://www-ikn.ist.hokudai.ac.jp/~arim/papers/arimura\\_stacs94.pdf](http://www-ikn.ist.hokudai.ac.jp/~arim/papers/arimura_stacs94.pdf)
  - $k$ -multiple minimal genelization ( $k$ -mmg) を多項式時間で求める

## 趣旨

description (pattern)  $\rightsquigarrow$  concept (language)  $\supseteq$  (given) strings  $S$

- 1 つの description は 1 つの concept を表現する
- 文字列の集合  $S$  が与えられる
- $S$  を網羅 (包含, covering) する言語を表現する description 及び、その concept を探索する
  - ただしそのような concept の中で極小なもの

$$S \sqsubseteq L_n \sqsubseteq \dots \sqsubseteq L_1 \sqsubseteq L_0$$

# Agenda

1. 汎化システム (Generalization System; GS)
  - 正規パターン言語での例
2. 言語の帰納的推論
  - 正規パターン言語での例
3. multiple description
4.  $k$ -mmg の構成アルゴリズム

## 汎化システム (Generalization System; GS)

一種の言語を構成する系で、以下で構成される

- description の全体集合  $D$
- $D$  上の半順序  $\preceq$
- 最大限  $\top \in D$
- 極小元を object と呼ぶ

# 汎化

$$p \preceq q$$

- $q$  は  $p$  の generalization (汎化)
  - $p$  から  $q$  への構成を generalize という
- $p$  は  $q$  の instance
  - $q$  から  $p$  への構成を refine という

## Concept by description

次を  $p$  で表現される concept という

$$L(p) = \{q \preceq p : q \text{ is object} \}$$

## 趣旨

concept は言語の一般化概念

- description  $\rightarrow$  Pattern
- concept  $\rightarrow$  Language

Language における包含関係  $(L, \subseteq)$  を Pattern における  $(p, \preceq)$  の関係で特徴づけたい



## Prop.

$$p \preceq q \Rightarrow L(p) \subseteq L(q)$$

- $s \in L(p)$
- $\iff s \preceq p$  (定義)
- $\iff s \preceq q$  (推移律)
- $\iff s \in L(q)$  (定義)

## Reverse

$$p \preceq q \stackrel{?}{\Leftrightarrow} L(p) \subseteq L(q)$$

一般にこれは成立しない。

これがいつも成立するような GS を complete GS という。

## 正規パターン (Regular Pattern; RP)

- 大きさ 2 以上の有限アルファベット集合:  $\Sigma = \{0, 1, 2, \dots\}$ 
  - 文字列 (object):  $\Sigma^+$
  - 空文字列:  $\Sigma^0 = \{\epsilon\}$
- 変数の無限集合  $X = \{x, y, z, \dots\}$
- パターンとは  $(\Sigma \cup X)^+$  で表現される列
- **正規パターン**: 一つの変数が高々一度出現するパターン
  - e.g.  $0x01y0$

## RP 上の $\preceq$

ある代入によって  $q \mapsto p$  となる関係を

$$p \preceq q$$

で定める

**代入** RP 中の一つの変数を別な RP で置き換えることによる RP から RP への順同型写像 (変数はかぶらないようにする)

- 消去可能パターン: 特別に空列の代入を許す (erasing)
- 消去不能パターン: 許さないもの (non-erasing)

## 代入の例

- $0x01z00 \preceq 0x01y0$
- $0x010 \preceq 0x01y0$  (erasing)

## 自明な代入

- 代入  $\{x := y\}$  (変数名の置き換え)
- 代入  $\{x := yz\}$  (erasing)

### 同値関係

$$p \preceq q \wedge p \succeq q \iff p \equiv q$$

変数名の置き換え、消去可能なら erasing は同値なパターンに写す

- $0x01 \equiv 0y01$
- $0x01 \equiv 0yz01$

正規パターンについてはこれの商集合をとることにする

## 標準形

左から  $i$  番目に出現する変数を  $x_i$  とリネームする

$$\blacksquare x_1 w_1 x_2 \dots x_n w_n x_{n+1}$$

- $x_i \in X$
- $w_i \in \Sigma^+$  (消去可能)
- $w_i \in \Sigma^*$  (消去不能)

商集合の代表元だと考える

## パターンの作る言語

- $L(0x01y0) = \{0x01y0 : x \in \Sigma^+, y \in \Sigma^+\}$
- $L(0x01y0) = \{0x01y0 : x \in \Sigma^*, y \in \Sigma^*\}$  (erasing pattern language)

### パターンにおける object

- $\preceq$  の最小元を object といったが
- RP においては明らかに  $\Sigma^+ (\Sigma^*)$  のこと
  - 代入を繰り返してできるもの



## completeness of RP language

- $p \preceq q \Rightarrow L(p) \subseteq L(q)$
- $p \preceq q \Leftarrow^? L(p) \subseteq L(q)$ 
  - $|\Sigma| > 2$  の時、これは成り立つ
  - $|\Sigma| = 2$  のときの反例

## 本スライドの趣旨

有限の object (文字列) 集合  $S$  が与えられたときに、 $S$  はどの言語から来たかを推論したい

すなわち、

$$S \mapsto p \quad \text{s.t.} \quad S \subseteq L(p)$$

■  $p \in D$  が  $S$  の covering である  $\iff S \subseteq L(p)$

## 推論

先の命題を満たすだけなら 自明な言語 がある

$$\forall S. S \subseteq L(T)$$

- $T$  は RP なら変数一つからなるパターン
  - これは嬉しくないだろう

## 推論

推論の“良さ”として言語の大きさによって定める

$$\min_p L(p) \text{ s.t. } S \subseteq L(p)$$

### 言語の大きさ

- $(D, \subseteq)$  によって言語の大小を比較する
- $p \subseteq q$  のとき、 $L(p)$  は  $L(q)$  より小さい
  - すなわち包含関係 (半順序) で極小となる言語

## 正提示からの帰納的推論 (Gold による形式化)

言語族  $\mathcal{L} = \{L(p) : p \in D\}$  (e.g. 正規パターン言語全体) について

- 言語族の任意の言語  $L(p)$  の元からなる任意のただし要素はすべて異なる無限列  $\sigma = (s_1, s_2, \dots)$  を正提示といい  $\sigma$  の頭  $n$  個を断片  $\sigma[n]$  という
- 推論アルゴリズム  $M$  とは
  - $M : \sigma[n] \mapsto (p_n \in D)$
  - $\exists N. \forall n > N. p_n = p$  となるもの
- 推論アルゴリズムが存在する言語族を推論可能な族だという

## 正規パターンは推論可能である

Prop

$p \preceq q$  のとき

- 消去可能パターン:  $\text{size}(p) \geq \text{size}(q)$  (アルファベットの数)
- 消去不能パターン:  $|p| \geq |q|$

## 正規パターンは推論可能である

object  $s$  が与えられた時、 $s$  の汎化する  $p$  ( $s \preceq p$ ) の size は  $s$  の size より小さい

- $S \subseteq L(p) \iff \forall i. s_i \preceq p$
- $\iff \min \text{size}(s_i) \geq p$

ある size 以下の正規パターンというのは有限しかない

- 消去可能で size  $n$  の最長のパターン
  - $x_1 a_1 x_2 \dots a_n x_{n+1}$

## 正規パターンは推論可能である

$S \subseteq L(q)$  となる  $q$  は有限通りしかないから全て試せばいい (部分点: 30 点)



# minl

- $S$  の covering であって言語が極小となる  $p$  を minimal common generalization (mcg) という
- mcg が作る言語を minimal language (minl) という

## 正規パターンの minl

例

- object 集合  $S$ 
  - 000111
  - 110111
  - 10011
  - 000100
- 直感: infix に 01 が出現する言語
  - $p = x01y$

最長共通部分列を取ればよさそう

## 正規パターンの minl

$S$  の最長共通部分列が  $a_1 a_2 \dots a_n$  なら

- $p = x_1 a_1 x_2 a_2 \dots a_n x_{n+1}$
- それぞれの変数について潰せたら潰す
  - $S \subseteq^? L(\{x_i := \epsilon\} p)$

## multiple description (和言語)

複数の description の和をとって高い表現力を得る

- $P = \{p_1, \dots, p_k\}$ 
  - $|P| \leq k$  の場合を特に  $k$ -multiple description という
  - description 全体  $D$  に対して  $k$ -multiple 全体を  $D^k$  と書く
- $L(P) = \cup_i L(p_i)$

## 汎化関係

$p \preceq q \Rightarrow L(p) \subseteq L(q)$  に相当する  $P$  の汎化関係  $\sqsubseteq$  を次で定める

$$P \sqsubseteq Q \iff \forall p \in P. \exists q \in Q. \Rightarrow L(p) \subseteq L(q)$$

- $P \sqsubseteq Q \Rightarrow L(P) \subseteq L(Q)$
- $P \sqsubseteq Q \Leftarrow L(P) \subseteq L(Q)$  **not** hold (even if complete)

## multiple description を用いた推論

object の有限集合  $S$  から

- $S \subseteq L(P)$  — covering
- 汎化  $(D^k, \sqsubseteq)$  において極小

を満たす  $P \in D^k$  を推論したい

- このような  $P$  を minimal multiple generalization (mmg) という

## 自明な multiple

$S$  に対して  $P = S$  そのものは

- $S \subseteq L(P)$  — 等しい
- $\forall Q (\neq S). Q \not\subseteq P$  — 極小

## $k$ -mmg

$P$  の良さとして  $P$  自体の単純さを加味する

すなわち  $k$ -multiple における mmg ( $k$ -mmg) の推論を考える



## 例 ( $k=2$ )

- object 集合  $S$ 
  - 000111
  - 010111
  - 100111
  - 000100
- 2-mmig として  $\{0001xy, xy0111\}$  など
- $\{xy01zw\}$  は 2-multiple であるが極小ではない

## To $k$ -mmg

$k$ -mmg を求めるのに手がかりとなる性質を述べていく

1. reduced  $k$ -multiple
2. tightest
3. division

## reduced $k$ -multiple

$S$  の covering となっている  $k$ -multiple  $P$  について

$$\forall Q \subset P. Q \text{ is not covering}$$

このとき  $P$  は reduced だという

「 $P$  の中に不要な  $p$  が含まれていないこと」

Prop.

$k$ -mmg ならば reduced である ( $\because Q \subset P \Rightarrow Q \sqsubseteq P$ )

- $S$  の reduced covering  $k$ -multiple は高々有限

## tightest $k$ -multiple

$P$  が  $S$  の tightest covering であるとは

$$\forall p \in P. p \text{ is mcg of } S \setminus L(P \setminus p)$$

「 $p$  は、 $p$  以外で cover していない文字列すべての極小共通汎化になっている。」

- tightest ならば reduced

## Theorem 4.1

$P$  が  $S$  の reduced covering でかつ、 $|P| = k$  ならば、

$$P \text{ is tightest} \iff P \text{ is } k\text{-mmg}$$

## 戦略

- $P = \{\top\}$  から始める (これは tightest)
- $|P| < k$  の間
  - tightest な  $P'$  でかつ  $|P| < |P'| \leq k$  を作る
  - $P \leftarrow P'$
- 大きさ  $k$  の  $P$  を得る

得られる  $P$  は  $k$ -mmg になっていることが保証される

- 大きさを調整しながら  $P$  からそれより大きな  $P'$  を作る必要がある

## $k$ -division

$S$  の covering である description  $p$  の  $k$ -division とは次のような multiple  $P$  のこと

- $P \sqsubset \{p\}$
- $1 < |P| \leq k$
- $S \subseteq L(P)$

$k$ -division は必ずしも存在しない

- 存在するとき、( $S$  に対する)  $p$  は  $k$ -divisible であるという

## $k$ -division 例

- $S = \{01, 12, 20\}$
- $p = xy$
- 3-division として  $S$  そのものがある
- 2-division は存在しない



## Theorem 4.2

$S$  の reduced covering  $k$ -multiple  $P$  について

$P$  is  $k$ -mmg  $\iff$

- $P$  is tightest and
- $\forall p \in P$ .  $p$  is **not**  $\delta k$ -divisible
  - where  $\delta k = k - |P| + 1$

## 戦略 (続き)

- $\delta k$ -division に従って  $P$  を大きくする
  - divisible でなくなった時
  - $k$ -mmg が保証される

## 手続き mmg のアルゴリズム: 入力 $(k, S)$

- $P \leftarrow \text{tightestCovering}(\{\top\}, S)$
- $\delta k \leftarrow k - 1$
- while  $\delta k \geq 1$  and  $\exists p \in P$ .  $p$  is  $\delta k$ -divisible to  $S \setminus L(P \setminus p)$ 
  - $p \leftarrow \delta k$ -divisible description in  $P$
  - $S' = S \setminus L(P \setminus p)$
  - $\Delta P = \delta k$ -division of  $(S', p)$
  - $P \leftarrow P \setminus \{p\} \cup \text{tightestCovering}(\Delta P, S')$
  - $\delta k = k - |P|$
- $\text{tightestCovering}(P, S)$  は
  - $S$  とその covering  $P$  を取って
  - tightest covering を返す ( $P$  から構成する)
  - ただし  $P$  と大きさは同じ

$$tc(P) = tightestCovering(S, P)$$

tightest とは

- any  $p \in tc(P)$ .  $p$  が  $S \setminus L(tc(P) \setminus p)$  の極小共通汎化になっていること

であった。

今、 $P$  が  $S$  の covering であるから covering を保ったまま 貪欲に refine すればよい (generalization の逆)

- loop
  - $p \leftarrow P$
  - assert  $q \preceq p$  and  $(S \setminus L(P \setminus p)) \subseteq L(q)$
  - $P \leftarrow P \setminus \{p\} \cup \{q\}$

## refine operator (近傍)

basic assign

- $\{x := a\} \ (a \in \Sigma)$
- $\{x := yz\}$

貪欲に refinement を探すのに上の二つを取れば十分

## 論文に載ってる例

- S

- 0000
- 00000
- 0002
- 0002222
- 00122
- 002222
- 002222
- 1211
- 112111
- 2221

- $\Sigma = \{0, 1, 2\}$

- $k = 4$

## 補遺

$|\Sigma| = 2$  の消去可能正規パターン言語の汎化システムは完全ではない

- $\Sigma = \{0, 1\}$
- $p = x_1 0 1 x_2 0 x_3$
- $q = x_1 0 x_2 1 0 x_3$

$L(p) = L(q)$  であるが、 $p \not\leq q$  かつ  $p \not\geq q$  である