数理解析研究所講究録
992巻 1997年 36-41

決定性有限メモリーオートマトンの学習可能性

坂本 比呂志
hiroshi@i.kyushu-u.ac.jp
九州大学システム情報科学研究科

Thomas Zeugmann
thomas@i.kyushu-u.ac.jp
九州大学システム情報科学研究科

要旨

有限メモリーオートマトンは，通常の有限オートマトンを，無数の要素を持つアルファベット上に拡張することによって得られる，文字列受理系である．本研究では，有限メモリーオートマトンの部分族である，単純決定性有限メモリーオートマトンの族が，所属性質問と等価性質問によって多項式時間学習可能であることを示す．

1 準備

ここでは，基本的な定義を与える．文字の集合をアルファベットという．自然数全体の集合を N と表すとき，$\hat{\Sigma} = \{a_i | i \in \mathrm{N}\}$ を加算無限なアルファベットという．Σ で，$\hat{\Sigma}$ の有限部分集合を表す．$\hat{\Sigma}$ 上の文字列とは，$w = w_1 w_2 \cdots w_n$ $(w_i \in \hat{\Sigma})$ のことである．$\hat{\Sigma}$ 上の文字列全体の集合を $\hat{\Sigma}^*$ と表し，$\hat{\Sigma}^+$ で，$\hat{\Sigma}^* \backslash \{\varepsilon\}$ を表す．ただし，ε は空の文字列である．いま，$\hat{\Sigma}$ に含まれない特別な文字 $\#$ を導入し，$\hat{\Sigma} \cup \{\#\}$ 上の文字列 w が $\hat{\Sigma}$ のすべての要素を高々一つしか含まないとき，w を $\Sigma \cup \{\#\}$ 上のアサイメントという．$w \in \hat{\Sigma}^*$ に対して，$|w|$ でその長さを表す．$[w]$ で w に含まれる文字の集合を表す．また，$w[i]$ で，w の i 番目の文字を表す．集合 S に対して，$||S||$ で S の要素数を表す．

定義 1 *[6]* 有限メモリーオートマトンとは，$A = \langle S, u, q_0, \rho, \mu, F \rangle$ のことである．S は有限集合であり，その要素を状態という．u は，$\hat{\Sigma} \cup \{\#\}$ 上のアサイメントである．$q_0 \in S$ は，初期状態である．ρ は，$S \times \{1, 2, \ldots, k\}$ の部分集合である $(k = |u|)$．μ は，$S \times \{1, 2, \ldots, k\} \times S$ の部分集合である．$F \subseteq S$ の要素は最終状態という．

$A = \langle S, u, q_0, \rho, \mu, F \rangle$ の動作は，つぎのように定義される．(p, u_i) を現在の状態と現在のアサイメントの組とするとき，(p, u_i) を現在の計算状況という．このとき，ある入力文字 $a \in \hat{\Sigma}$ に対して，

1. $a = u_i[j]$ である場合，もし $(p, j, q) \in \mu$ であるならば状態を q に変える．

2. $a \notin [u_i]$ である場合，もし $\rho(p) = j$ でかつ $(p, j, q) \in \mu$ であるならば $u_i[j] := a$ として状態を q に変える．

つぎの計算状況を (q, u_{i+1}) と表すと入力文字 a に対して，$(p, u_i) \vdash^a (q, u_{i+1})$ という関係を定義する．(q_0, u) を初期計算状況といい，$p \in F$ に対して，(p, u_i) を受理計算状況という．$w \in \hat{\Sigma}^*$ に対して，$(q_0, u) \vdash^* (p, u')$ かつ $p \in F$ であるとき，A は，w を受理するという．ただし，$\vdash^*$ は，$\vdash$ の反射的推移閉包である．

定義 2 *[6]* 有限メモリーオートマトン $A = \langle S, u, q_0, \rho, \mu, F \rangle$ について，ρ が写像であり，すべての $p \in S$ と $i \in \{1, 2, \ldots, k\}$ に対してただひとつの $q \in S$ が存在して，$(p, i, q) \in \mu$ であるとき，A は，決定性であるという．

定義 3 有限メモリーオートマトン $A = \langle S, u, q_0, \rho, \mu, F \rangle$ について，$u \in \{\#\}^*$ であるとき，A は，単純であるという．

以下では，単純決定性有限メモリーオートマトンを dfma#，決定性有限オートマトンを dfa と表す．

2 決定問題

ここでは，学習アルゴリズムの構築のために，dfa と dfma# に関する二つの決定問題が，いずれも可解であることを示す．

2.1 有限オートマトンに関する決定問題

$M = \langle Q, \Sigma, q_0, \delta, F \rangle$ を，dfa とする．$\ell : \Sigma \mapsto \Sigma$ を，Σ 上の自己同型写像とする．$\ell(L(M)) = \cup_{w \in L(M)} \ell(w)$ と定義する．すべての ℓ に対して，$L(M) = \ell(L(M))$ であるとき，M は，自己同型写像について閉じているという．

定義 4 M と $a, b \in \Sigma$ に対して，ある *dfa* $M_{a|b} = \langle Q, \Sigma, q_0, \delta_{a|b}, F \rangle$ を以下のように定義する．

$$\delta_{a|b}(p, c) = \begin{cases} \delta(p, a), & c = b \\ \delta(p, b), & c = a \\ \delta(p, c), & c \in \Sigma \backslash \{a, b\} \end{cases}$$

定義 5 M に対して，$M^1 = \{M_{a|b} | a, b \in \Sigma\}$ および $M^{K+1} = \{M_{a|b} | M \in M^k, a, b \in \Sigma\}$ と定義する．

定義 6 Σ 上の代入の集合 Π_k $(k \geq 1)$ を次のように定義する．$\Pi_1 = \{a \mapsto b, b \mapsto a | a, b \in \Sigma\}$, $\Pi_{k+1} = \{\pi_i \pi_j | \pi_i \in \Pi_1, \pi_j \in \Pi_k\}$.

補題 1 すべての $k \geq 1$ に対して，つぎの条件は同等である．

1. $L(M) = L(M')$, $M' \in M^k$

2. $L(M) = \pi(L(M))$, $\pi \in \Pi_k$.

補題 2 すべての $k \geq 1$ に対して，つぎの条件は同等である．

1. $L(M) = L(M'),\ M' \in M^1$

2. $L(M) = L(M'),\ M' \in M^k$.

ここで，任意に与えられた dfa M_1, M_2 に対して，$L(M_1) = L(M_2)$ であるか否かを $k = max(size(M_1), size(M_2))$ に関する多項式時間で決定できるアルゴリズムが存在することに注意する．ただし，$size(M) = ||Q|| \cdot ||\Sigma||$ である．したがって，つぎの定理が成立する．

定理 1 *dfa M* が，アルファベット上の自己同型写像について閉じているか否かという問題は，多項式時間で決定可能である．さらに，ある $\ell : \Sigma \mapsto \Sigma$ に対して，$L(M) \neq \ell(L(M))$ であるならば，ある文字列 $w \in L(M) \oplus \ell(L(M))$ を，やはり多項式時間で発見することができる．ただし，$\oplus$ は排他的論理和を表す．

2.2 単純決定性有限メモリーオートマトンの等価性

dfma# A_1, A_2 に関する等価性とは，$\hat{\Sigma}$ 上で $L(A_1) = L(A_2)$ であるか否かという問題のことである．

定義 7 $\mathcal{A} = \langle S, \mathrm{u}, q_0, \rho, \mu, F \rangle$ を *dfma#* とする．$\Sigma = \{a_0, a_1, \ldots, a_k\}$ を $k \geq |u|$ であるような有限なアルファベットとする．文字列 $w \in \hat{\Sigma}^+$ に対して，集合 $w_\Sigma^A \subseteq \Sigma^{|w|}$ を以下のように定義する．

$$\begin{aligned} a_\Sigma^A &= \Sigma \quad a \in \hat{\Sigma} \\ (wa)_\Sigma^A &= \bigcup_{w' \in (w)_\Sigma^A} \{w'a'\} \begin{cases} a' = u_{w'}[i], & a = u_w[i] \\ a' \in \Sigma \backslash [u_{w'}], & a \notin [u_w] \end{cases} \end{aligned}$$

ただし，u_w は，A が w を読み終えたときのアサイメンとを表す．

補題 3 すべての $w \in \hat{\Sigma}^+$ と $w' \in w_\Sigma^A$ に対して，$A(w) = A(w')$ である．ただし，$A(w) = A(w')$ であるとは，$(q_0, u) \vdash^w (p, u_i)$ かつ $(q_0, u) \vdash^{w'} (p, u_j)$ であることを表す．

補題 4 A と B を *dfma#* とし，Σ を $||\Sigma|| > |u^A| + |u^B|$ であるような $\hat{\Sigma}$ の有限部分集合とする．すべての $w \in \hat{\Sigma}^+$ に対して，$w_\Sigma^A \cap w_\Sigma^B \neq \emptyset$ である．

上の二つの補題により，つぎの定理が成立する．

定理 2 A と B を *dfma#* とし，Σ を $||\Sigma|| > |u^A| + |u^B|$ であるような $\hat{\Sigma}$ の有限部分集合とする．このときつぎの条件は同等である．

1. $\hat{\Sigma}$ 上で $L(A) = L(B)$

2. Σ 上で $L(A) = L(B)$.

任意の dfma A と任意の有限な $\Sigma \subset \hat{\Sigma}$ に対して，$L = \Sigma^* \cap L(A)$ は正則であり，$L = L(M)$ であるオートマトンは計算可能である．したがって，つぎの系を得る．

系 1 *dfma#* に関する等価性問題は決定可能である．

3 dfam# の学習

3.1 問題設定

任意の dfma# A が学習対象である．あるアルゴリズムが存在して，その出力 B に対して，$L(A) = L(B)$ となるとき，dfma# の族は学習可能であるという．また，そのアルゴリズムの計算時間が，学習対象 A のサイズの多項式で抑えられるとき，dfma# の族は，多項式時間学習可能であるという．ここで，アルゴリズムには，つぎの 2 種類の質問を使用することが許されているものとする．

1. 所属性質問：任意の文字列 $w \in \hat{\Sigma}^*$ に対して，$w \in L(A)$ であるか否か．

2. 等価性質問：任意の dfma# B に対して，$L(A) = L(B)$ であるか否か．もし，$L(A) \neq L(B)$ であるならば，ある文字列 $w \in L(A) \oplus L(B)$ を受けとる．この文字列を A の B に対する反例と呼ぶ．

アルゴリズムが，上の 2 種類の質問を用いて A を多項式時間学習するとき，dfma# の族は，所属性質問と等価性質問から，多項式時間学習可能であるという．

3.2 学習アルゴリズム

以下に，学習アルゴリズムの概要を示す．A_* を学習対象とする．E をそれまでに受けとった反例の集合とし，$\Sigma = \cup_{w \in E}[w]$ と定義する．

Step1. A に関する所属性質問使って，E と Σ から，無矛盾でかつ閉じている観測表 T を構成する．観測表の概念については，[1] を参照．
Step2. T から，T と無矛盾な dfa M を構成する．すべての $\ell : \Sigma \mapsto \Sigma$ に対して，$L(M) = \ell(L(M))$ であるか否かを計算する．$L(M) \neq \ell(L(M))$ であるならば，$w \in L(M) \oplus \ell(L(M))$ を発見し，$E := E \cup \{w\}$ として Step1 に戻る．
Step3. サブルーチン **Cons** によって，$L(M) = \Sigma^* \cap L(A)$ であるような dfma# A を計算する．
Step4. サブルーチン **Mini** によって，$\Sigma^* \cap L(A) = \Sigma^* \cap L(B)$ であるようなアサイメントの長さが最小な dfma# B を計算する．
Step5. 等価性質問によって，$L(B) = L(A_*)$ であるならば，B を出力して停止．$L(B) \neq L(A_*)$ ならば，$E := E \cup \{w\}, \Sigma = \cup_{w \in E}[w]$ として Step1 に戻る．

Step3 において，ある dfa $M = \langle Q, \Sigma, q_0, \delta, F \rangle$ に対して，サブルーチン **Cons** は，つぎのように動作する．$S := Q$, $u := \#^k$ $(k = ||\Sigma||)$, $q_0^A := q_0$, $(p, a_i, q) \in \delta \Leftrightarrow^{\text{def}} (p, i, q) \in \mu$, $F^A := F$. さらに，すべての $1 \leq i \leq k$ に対して，$\rho(p) = i \Leftrightarrow^{\text{def}} \rho(p) \notin \rho$ であり，ある文字列 $w \in \Sigma \backslash \{a_i\}$ が存在して，$\delta(q_0, w) = p$. 以上のように定義された S, u, q_0^A, ρ, μ および F^A に対して，**Cons** は，dfma# $A = \langle S, u, q_0^A, \rho, \mu, F^A \rangle$ を出力する．

Mini の動作を説明するために，dfa の状態 p と q に対して，関係 $p \equiv_\ell q$ をつぎのように定義する．

定義 8 *dfa* $M = \langle Q, \Sigma, q_0, \delta, F\rangle$ に対して，$W_p = \{w \in \Sigma^* | \delta(q_0, w) = p\}$ と定義する．$p, q \in Q$ に対して，$p \equiv_\ell q \Leftrightarrow^{def}$ ある自己同型写像 $\ell : \Sigma \mapsto \Sigma$ が存在して $W_q = \ell(W_p)$ である．ただし，$\ell(W_p) = \cup_{w \in W_p}\{\ell(w)\}$ である．

Step4 において，ある dfa $M = \langle Q, \Sigma, q_0, \delta, F\rangle$ と dfma# $A = \langle S, u, q_0^A, \rho, \mu, F^A\rangle$ に対して，サブルーチン **Mini** は，つぎのように動作する．まず，すべての $p, q \in Q$ に対して，$p \equiv_\ell q$ であるか否かを判定する．もし，そうでなければ A を出力する．loop： $\rho(p) = k$ であるようなすべての $p \in S$ に対して，ある $q \in S$ が存在して，$(p, k, q'), (p, i, q) \in \mu$, $i < k$ かつ $q' \equiv_\ell q$ であるか否かを判定する．もし，$q' \not\equiv_\ell q$ ならば，A を出力する．そうでなければ，$\rho(p) = i$, $u := \#^{k-1}$ として loop にもどる．

3.3 アルゴリズムの解析

補題 5 *dfa* M を $L = L(M)$ であるもののうち，状態数最小のものとする．このとき，M の任意の状態 p, q に対してつぎのうちどちらかが成立する．

1. $p \equiv_\ell q$
2. $W_q \cap \ell(W_p) = \emptyset$.

Angluin [1] の結果によって，Step2 で得られる dfa は，状態数最小のものである．したがって，上の補題からつぎの補題を導く．

補題 6 *Step1* で得られる任意の *dfa* に対して，関係 $\equiv_\ell$ は，判定可能である．

さらに，関係 $\equiv_\ell$ が成立しないとき，その反例を発見することができる．また，Σ は有限であるので，その反例もまた有限個しか存在しない．したがって，有限個の反例を発見することによって，Step3 に進むことができる．

補題 7 任意の *dfaM* に対して，ある *dfma#* A が存在して，$L(M) = \Sigma^* \cap L(A)$ であるならば，**Cons** の出力は，その中の一つである．

補題 8 **Mini** の出力は，アサイメントの長さが最小のものである．

前節の定理 2 によって，Σ のサイズが十分に大きいとき，dfma# の等価性は dfa の等価性と同等であることがわかった．したがって，上の 2 つの補題によって，**Mini** で出力される dfma# は，いつかは目標とされる dfma# と同等となることがわかる．以上により，つぎの定理を得る．

定理 3 *dfma#* の族は，所属性と等価性質問によって学習可能である．

つぎに，計算時間を評価する．dfa M に対して，$L(M) = \ell(L(M))$ であるか否かと，$p \equiv_\ell q$ であるか否かを多項式時間で判定できることを示せば十分である．$L(M) = \ell(L(M))$ に対しては，前節の定理 1 によって多項式時間で判定可能である．さらに，

補題 9 *dfa* $M = \langle Q, \Sigma, q_0, \delta, F \rangle$ を状態数最小のものとするとき，任意の $p, q \in Q$ に対して，$p \equiv_\ell q$ であるか否かは，$||Q||$ と $||\Sigma||$ の多項式時間で判定可能である．

したがって，アルゴリズムの Step1 から Step5 に到達するためにかかる時間は，k と n に関する多項式時間である．ただし，k は A_* のアサイメントの長さであり，n は，$||\Sigma|| = 2k+1$ に対して，$L(M) = \Sigma^* \cap L(A)$ を満たすオートマトンの最小の状態数である．さらに，定理 2 によって，高々 $2k$ 個の反例を受けとることによって，目標とする dfma# と等価な dfma# を計算することができる．したがって，計算時間全体は，k, n および m に関する多項式時間で抑えられる．ただし，m は，受けとった最長の反例の長さである．以上により，本研究の結果として定理 4 を得る．

定理 4 *dfma#* の族は，所属性と等価性質問によって多項式時間学習可能である．

参考文献

[1] D. Angluin. *Learning regular sets from queries and counterexamples.* Information and Computation, 75:87-106, 1987.

[2] D. Angluin. *Queries and concept learning.* Machine Learning, 2:319-342, 1988.

[3] W.I. Gasarch and C.H. Smith. Learning via queries. Journal of the ACM, 39(3):649-674, 1992.

[4] R. Gavaldà. *On the power of equivalence queries.* Proceedings of the 1st European Conference on Computational Learning Theory, Royal Holloway University, London, 1993.

[5] J. E. Hopcroft and J. D. Ullman. *Introduction to Automata Theory, Languages, and Computation.* Addison-Wesley, 1979.

[6] M. Kaminski and N. Francez. *Finite-memory automata.* Theoretical Computer Science, 134:329-363, 1994.

[7] E. Shapiro, *Algorithmic program debugging*, Cambridge, MA: MIT Press, 1983.

[8] H. U. Simon. *Learning decision lists and trees with equivalence-queries.* Proceedings of the 2nd European Conference on Computational Learning Theory, Barcelona, Spain, 1995.

[9] O. Watanabe. *A formal study of learning via queries.* Proceedings of the 17th International Colloquium on Automata, Languages and Programming, pp.139-152, 1990.