Polynomial Time Learnability of Deterministic Right Linear Translations

Sugimoto, Noriko
Department of Artificial Intelligence Kyushu Institute of Technology

Ishizaka, Hiroki
Department of Artificial Intelligence Kyushu Institute of Technology

出版情報：RIFIS Technical Report. 120, 1995-10-27. Research Institute of Fundamental Information Science, Kyushu University

RIFIS-TR-CS-120

RIFIS Technical Report

Polynomial Time Learnability of Deterministic Right Linear Translations

Noriko Sugimoto and Hiroki Ishizaka

October 27, 1995

Research Institute of Fundamental Information Science
Kyushu University 33
Fukuoka 812, Japan
E-mail: sugimoto@ai.kyutech.ac.jp
Phone: 0948-29-7630

Polynomial Time Learnability of Deterministic Right Linear Translations

Noriko Sugimoto Hiroki Ishizaka
e-mail: {sugimoto, ishizaka}@ai.kyutech.ac.jp

Department of Artificial Intelligence
Kyushu Institute of Technology
kawazu 680-4, Iizuka 820, Japan

TEL: +81-948-29-7630, FAX: +81-948-7601

Abstract

This paper is concerned with the problem of learning translations from minimally adequate teacher. A translation considered here is a binary relation over strings and defined by an elementary formal system (EFS, for short) in a special form. An EFS consists of several predicates, in general, but the defined translation is directly characterized by only one designated predicate in them. This means that every other predicates, which are indirectly necessary for defining the translation, are never observed in the interaction between a teacher and a learner. The main problem investigated in this paper is how to inventing such unobserved predicates and to complete learning. The presented algorithm learns successfully the target EFS inventing such auxiliary predicates via membership and equivalence queries in polynomial time.

連絡先:
杉本 典子 (すぎもと のりこ)
〒 820 飯塚市大字川津 680-4
九州工業大学 情報工学部 知能情報工学教室
TEL: +81-948-29-7630, FAX: +81-948-7601
EMAIL: sugimoto@ai.kyutech.ac.jp

1 Introduction

We consider the problem of learning translations using membership and equivalence queries, that is, from a minimally adequate teacher. A translation considered here is a binary relation over strings and defined by an elementary formal system (EFS, for short) in a special form, called a translation EFS (TEFS, for short).

The EFS's are well known to be flexible enough to define various classes of languages in Chomsky hierarchy [5]. Furthermore, since the EFS's are logic programs over strings, they can easily define various relations over strings. The TEFS's are one of such EFS's, which are especially tuned to define binary relations over strings, that is, translations. Sugimoto introduced several classes of translations defined by TEFS's and gave some properties of them [15]. In this paper, we focus on the learnability of a particular class of translations defined by some restricted TEFS's, called deterministic right linear TEFS's (DR-TEFS, for short). The class of translations is a proper super-class of those defined by sequential machines and a proper sub-class of those defined by sequential transducers.

The learning model investigated in this paper is polynomial time exact learning from a minimally adequate teacher. In general, each translation is defined as a partial Herbrand model of a DR-TEFS, which consists of all the elements with one designated predicate symbol in the model. The minimally adequate teacher assumed in this paper can answer questions about the translation, that is, the partial Herbrand model. This means that the target DR-TEFS might include predicates never observed in the interaction between the teacher and the learner. Thus, in order to learn the target DR-TEFS, the learner also has to invent such necessary but unobserved predicates for itself.

This problem of learning both concepts and languages for describing the concepts, in other words "inventing theoretical terms", is one of the most important and difficult theme in machine learning [6, 8, 9, 10, 11]. Ishizaka gave algorithms which work well in such framework [8, 9]. The presented algorithm is based on his algorithm. That is, the algorithm learns not only a target TEFS but also the predicates which are necessary for the TEFS but never observed in the interaction between a teacher and a learner. We show that the algorithm learns any translations definable by a DR-TEFS from a minimally adequate teacher in polynomial time.

2 Preliminaries

In this section, we present basic definitions needed in the following sections. We define elementary formal systems according to [5, 13, 14, 16] and translations according to [1, 2].

2.1 TEFS

Let Σ, X and Π be mutually disjoint sets. We assume that Σ is finite. We refer to each element of Σ as a *constant symbol*, to each element of X as a *variable*, and to each element of Π as a *predicate symbol*. Each predicate symbol is associated with a non-negative integer called its *arity*. In what follows, variables are denoted by x, y. For a finite set A, we denote the set of all finite strings of symbols from A by A^*. A *term* is an element of $(\Sigma \cup X)^*$. A term is said to be *ground* if it is an element of Σ^*. The length of a term π is denoted by $|\pi|$. An *atomic formula* (*atom*, for short) is of the form $p(\pi_1, \pi_2, \ldots, \pi_n)$, where p is a predicate symbol with arity n and each π_i is a term ($1 \leq i \leq n$). The *size* of an atom $\alpha = p(\pi_1, \ldots, \pi_n)$, denoted by $size(\alpha)$, is defined $size(\alpha) = |\pi_1| + \cdots + |\pi_n|$. An atom $p(\pi_1, \pi_2, \ldots, \pi_n)$ is said to be *ground* if all $\pi_1, \pi_2, \ldots, \pi_n$ are ground. A *definite clause* (*clause*, for short) is of the form

$A \leftarrow B_1, \ldots, B_n$ $(n \geq 0)$, where $A, B_1, \ldots, B_n$ are atoms. The atom A is called the *head* and the sequence $B_1, \ldots, B_n$ of atoms is called the *body* of the clause. For a clause C, the head of C is denoted by $head(C)$. A *goal clause* (*goal*, for short) is of the form $\leftarrow B_1, \ldots, B_n$ $(n \geq 0)$ and the goal with $n = 0$ is called the *empty goal*. A *substitution* is a finite set of the form $\{x_1/\pi_1, \ldots, x_n/\pi_n\}$, where $x_1, \ldots, x_n$ are distinct variables and each π_i is a term distinct from x_i $(0 \leq i \leq n)$. We refer to either a term, an atom or a clause as an *expression*. Let E be an expression. Then, for a substitution $\theta = \{x_1/\pi_1, \ldots, x_n/\pi_n\}$, $E\theta$, called an *instance* of E, is the expression obtained from E by simultaneously replacing each occurrence of the variable x_i in E by the term π_i $(i = 1, \ldots, n)$. Let α and β be a pair of expressions and θ be a substitution. If $\alpha\theta = \beta\theta$ then we say that θ is a *unifier* of α and β. When there exists a unifier of α and β, we say that α and β are *unifiable*.

An *elementary formal system* (*EFS*, for short) is a finite set of clauses [14]. The set of all predicate symbols which occur in Γ is denoted by Π_Γ. We define a *translation EFS* (*TEFS*, for short) as a EFS with at least one predicate symbol with arity 2. Let Γ be a TEFS, and $q_0 \in \Pi_\Gamma$ be a predicate symbol with arity 2.

A TEFS Γ is *right linear* if each clause of Γ is of one of the following forms:

1. $p(\varepsilon, \varepsilon) \leftarrow$,
2. $p(ax, by) \leftarrow q(x, y)$,

where $a, b \in \Sigma$ and ε is a term whose length is 0. A right linear TEFS Γ is said to be *deterministic* if, for each $p \in \Pi_\Gamma$ and each $a, b \in \Sigma$, Γ includes at most one clause whose head is of the form $p(ax, by)$. In what follows, we refer to a deterministic right linear TEFS as *DR-TEFS* for short.

For an EFS Γ and ground atom α, a *derivation tree* of α on Γ is a finite tree that satisfies the following conditions.

1. Each node of the tree is a ground atom.
2. The root node is α.
3. For each internal node α and its children $\beta_1, \ldots, \beta_n$ $(n \geq 1)$, $\alpha \leftarrow \beta_1, \ldots, \beta_n$ is a ground instance of a clause in Γ.

A *proof tree* of α on Γ is a derivation tree of α on Γ such that each leaf of the tree is a ground instance of a clause with empty body in Γ. We define

$$SS(\Gamma) = \{\alpha \mid \alpha \text{ is a ground atom and there exists a proof tree of } \alpha \text{ on } \Gamma\}.$$

If Γ is a DR-TEFS, since for any ground atom α, there exists at most one clause in Γ whose head and α are unifiable, then any possible derivation tree of α is unique and of the very simple form as in Figure 1.

$$\begin{array}{c} q_i(a_1a_2a_3 \ldots a_n, b_1b_2b_3 \ldots b_n) \\ | \\ q_{i_1}(a_2a_3 \ldots a_n, b_2b_3 \ldots b_n) \\ | \\ q_{i_2}(a_3 \ldots a_n, b_3 \ldots b_n) \\ \vdots \end{array}$$

Figure 1: A derivation tree on a DR-TEFS

In this paper, we assume some effective procedure which, for any given ground atom α and any given DR-TEFS Γ, constructs a proof tree of α on Γ if it exists. Such a procedure can be implemented based on SLD-resolution [4]. For a general EFS, the computational cost of the procedure is rather expensive. However, for a DR-TEFS, we can construct any proof tree very easily. In fact, for any given DR-TEFS Γ and given ground atom α, the problem whether there exists a proof tree of α on Γ is decidable in linear time in $size(\alpha)$.

2.2 Models and translations

The *Herbrand base*, denoted by $\mathcal{B}$, is the set of all ground atoms. For an EFS Γ, the *Herbrand base* of Γ, denoted by $\mathcal{B}(\Gamma)$, is the set of all ground atoms whose predicate symbols are in Π_Γ. An *interpretation* is a subset of $\mathcal{B}$.

Let $C = A \leftarrow B_1, \ldots, B_n$ $(n \geq 0)$ be a clause and M be an interpretation. A ground atom α is said to be *covered by C with respect to M* if there exists a substitution θ such that $\alpha = A\theta$ and $B_i\theta \in M$ for each i $(1 \leq i \leq n)$. The set of all ground atoms covered by C with respect to M is denoted by $C(M)$. A clause C is said to be *true* in M if $C(M) \subseteq M$, otherwise C is said to be *false*.

An *Herbrand model* of Γ is an interpretation which makes every clause in Γ true. We define

$$M(\Gamma) = \cap\{M \mid M \text{ is an Herbrand model of } \Gamma\}$$

then, $M(\Gamma)$ is also an Herbrand model of Γ, and called the *least Herbrand model* of Γ. Yamamoto [16] showed that $M(\Gamma) = SS(\Gamma)$ for any EFS. Thus, from the argument in the previous subsection, the membership of arbitrary ground atom α in $M(\Gamma)$ is decidable in linear time in $size(\alpha)$.

A *translation* is a subset of $\Sigma^* \times \Sigma^*$. For a TEFS Γ, we define

$$T(\Gamma, q_0) = \{(w_1, w_2) \in \Sigma^* \times \Sigma^* \mid q_0(w_1, w_2) \in M(\Gamma)\}.$$

A translation T is said to be *defined by* a TEFS Γ if $T = T(\Gamma, q_0)$ for some $q_0 \in \Pi_\Gamma$. The predicate symbol q_0 is called the *start symbol*. For a translation T, if there exists a TEFS Γ such that $T = T(\Gamma, q_0)$ then T is said to be *definable by* TEFS's. We refer to a translation which is definable by DR-TEFS's as a *DR-translation*.

For right linear TEFS's, we can obtain the following theorem.

Theorem 1 *For any translation T, T is definable by right linear TEFS's if and only if T is definable by DR-TEFS's.*

We can prove the above theorem along the same line of argument as in the proof of equivalence of deterministic finite automata and non-deterministic finite automata. Furthermore, from the definition, it can be easily shown that the class of translations defined by DR-TEFS's is a proper superclass of those of sequential machines and a proper subclass of those of sequential transducers [7]. A sequential machine defines a translation from one string to one string with same length. A sequential transducer can define a translation from one string to many strings with different length. On the other hand, a DR-TEFS defines a translation from one string to many strings with same length. Essentially, all of them define translations over regular languages.

2.3 Types of queries

The algorithm described in the following section is allowed to use two types of queries: membership queries and equivalence queries.

Let T be a target DR-translation. A *membership query* proposes a pair of strings $(w_1, w_2) \in \Sigma^* \times \Sigma^*$ and asks whether $(w_1, w_2) \in T$. The reply is *yes* or *no*. An *equivalence query* conjectures a DR-TEFS Γ and asks whether $T = T(\Gamma, q_0)$. The reply is either *yes* or *no*. If it is *no*, then a *counterexample* is also provided. A counterexample is a pair of strings (w_1, w_2) in the symmetric difference of T and $T(\Gamma, q_0)$. If $(w_1, w_2) \in T - T(\Gamma, q_0)$ then (w_1, w_2) is called a *positive* counterexample, and if $(w_1, w_2) \in T(\Gamma, q_0) - T$ then (w_1, w_2) is called *negative* counterexample. The choice of a counterexample is assumed to be arbitrary. According to Angulin [3], we call a teacher who answers equivalence queries and membership queries a *minimally adequate teacher*.

3 A learning algorithm

In this section, we show a learning algorithm for DR-translations based on Ishizaka's inference algorithm [8, 9]. In what follows, we fix a target translation T. Furthermore, we assume that a start symbol q_0 and Σ are fixed and known to the learning algorithm. A TEFS Γ such that $T = T(\Gamma, q_0)$ and $\Pi_\Gamma - \{q_0\}$ are intended to be constructed by the algorithm.

3.1 An extended model

The algorithm given in this section is essentially based on Shapiro's model inference theory [12]. Our setting, however, differs from Shapiro's in available information to the leaner. In our setting, the learner is assumed to be given only one predicate q_0 and the interpretation for it via interactions with the minimally adequate teacher. Hence, the learner has to generates necessary predicates except for q_0 and construct an appropriate interpretation for them. Our algorithm makes the construction according to the model extension technique given in [8].

A *predicate characterization* for a DR-TEFS Γ, denoted by CH_Γ, is a one to one mapping from Π_Γ to $\Sigma^* \times \Sigma^*$. Let T be a translation and CH_Γ be a predicate characterization for a DR-TEFS Γ. We define an *extended model of T with CH_Γ*, denoted by $I(T, CH_\Gamma)$, as follows.

$$I(T, CH_\Gamma) = \{q_i(w_1, w_2) \in \mathcal{B}(\Gamma) \mid CH_\Gamma(q_i) = (u, v) \text{ and } (uw_1, vw_2) \in T\}.$$

From the definition, we can obtain a model over $\mathcal{B}(\Gamma)$ defined by T and CH_Γ. However, it is nonsense that arbitrary model is produced from T. For a suitable extension of the model, $I(T, CH_\Gamma)$ should satisfy the following condition:

$$T(\Gamma, q_0) = T \iff M(\Gamma) = I(T, CH_\Gamma).$$

We show some conditions of CH_Γ satisfying the above condition. Let CH_Γ be a predicate characterization for a DR-TEFS Γ. The CH_Γ is said to be *consistent* if $CH_\Gamma(q_i) = (u, v)$ for any $q_i \in \Pi$, then there exists a derivation tree of $q_0(u, v)$ on S in which $q_i(\varepsilon, \varepsilon)$ appears.

Lemma 2 *For any DR-TEFS Γ, if a predicate characterization CH_Γ for Γ is consistent, then it holds that $M(\Gamma) = I(T(\Gamma, q_0), CH_\Gamma)$.*

Proof: From the uniqueness of the derivation tree on Γ and consistency of CH_Γ, for any $q_i \in \Pi_\Gamma$ and $w_1, w_2 \in \Sigma^*$, if $CH_\Gamma(q_i) = (u, v)$, then it holds that

$$q_i(w_1, w_2) \in M(\Gamma) \iff q_0(uw_1, vw_2) \in M(\Gamma).$$

Hence, it holds that

$$\begin{aligned} q_i(w_1, w_2) \in M(\Gamma) &\iff q_0(uw_1, vw_2) \in M(\Gamma) \\ &\iff (uw_1, vw_2) \in T(\Gamma, q_0) \\ &\iff q_i(w_1, w_2) \in I(T(\Gamma, q_0), CH_\Gamma). \end{aligned}$$

$\square$

Theorem 3 *Suppose CH_Γ for a DR-TEFS Γ is consistent and $CH_\Gamma(q_0) = (\varepsilon, \varepsilon)$. Then, for any translation T, $T(\Gamma, q_0) = T$ if and only if $M(\Gamma) = I(T, CH_\Gamma)$.*

Proof: Since the *only if* direction immediately follows from Lemma 2, it is sufficient to prove the *if* direction. For any $w_1, w_2 \in \Sigma^*$, it holds that

$$\begin{aligned} (w_1, w_2) \in T &\iff q_0(w_1, w_2) \in I(T, CH_\Gamma) \quad (\text{from } CH_\Gamma(q_0) = (\varepsilon, \varepsilon)) \\ &\iff q_0(w_1, w_2) \in M(\Gamma) \quad (\text{from the assumption}) \\ &\iff (w_1, w_2) \in T(\Gamma, q_0). \end{aligned}$$

$\square$

Therefore, for a target translation T, we should construct a DR-TEFS Γ and a consistent predicate characterization CH_Γ with $CH_\Gamma(q_0) = (\varepsilon, \varepsilon)$.

3.2 A leaning algorithm for DR-translations

We need to give some more definitions and properties of DR-TEFS's before explaing our algotihm.

We say that a clause C is *sufficient in* an interpretation M if, for any $\alpha \in M$, if α is unifiable with the head of C, then it holds that $A \in C(M)$. A clause C is said to be *insufficient in M* if C is not sufficient in M. A clause C is said to be *complete in M* if C is both true and sufficient in M. The following proposition is directly obtained from the definition.

Proposition 4 *Let M be an arbitrary interpretation. Then, the following two statements are equivalent.*

1. *A clause $q_i(ax, by) \leftarrow q_j(x, y)$ is complete in M.*
2. *For any $w_1, w_2 \in \Sigma^*$, $q_i(aw_1, bw_2) \in M$ if and only if $q_j(w_1, w_2) \in M$.*

For any DR-TEFS Γ, since there exists at most one clause which covers a ground atom α in $M(\Gamma)$, each clause in Γ is sufficient in $M(\Gamma)$. Thus, we have the following proposition.

Proposition 5 *For any DR-TEFS Γ, every clause in Γ is complete in $M(\Gamma)$.*

With Lemma 2, this implies the following proposition.

Proposition 6 *Let T be a translation and CH_Γ be a consistent predicate characterization for a DR-TEFS Γ. If $T(\Gamma, q_0) = T$ then every clause in Γ is complete in $I(T, CH_\Gamma)$.*

From the above proposition, if a hypothesis Γ has a clause which is not complete in an extended model $I(T, CH_\Gamma)$, then Γ is incorrect. Hence, such incomplete clauses must be removed from the hypothesis.

Algorithm 1 : A learning algorithm for DR-TEFS's
Given : A minimally adequate teacher for T.
Output : A DR-TEFS Γ such that $T(\Gamma, q_0) = T$.
Procedure:
    $\Gamma := \emptyset$; $CH_\Gamma := (q_0, (\varepsilon, \varepsilon))$; $State = 0$;
    repeat
        make an equivalence query with Γ;
        if the reply is a positive counterexample (w_1, w_2) **then**
            Let P be the proof tree of $q_0(w_1, w_2)$ on Γ;
            C :=**contradiction_backtracing**(P);
            $\Gamma := \Gamma - \{C\}$;
            C' :=**next_clause**(C);
            $\Gamma := \Gamma \cup \{C'\}$;
        if the reply is a negative counterexample (w_1, w_2) **then**
            α =**uncovered_atom**($q_0(w_1, w_2)$);
            C =**search_clause**(α);
            $\Gamma = \Gamma \cup \{C\}$;
    until the reply is *yes*;
    output Γ.

contradiction_backtracing :
Given: A minimally adequate teacher for T.
Input: A proof tree of an atom $q_i(au, bv)$ on Γ such that $q_i(au, bv) \in M(\Gamma)$ but $q_i(au, bv) \notin I(T, CH_\Gamma)$.
Output: A clause $C \in \Gamma$ which is false in $I(T, CH_\Gamma)$.
Procedure:
    let $q_j(u, v)$ be the child of $q_i(au, bv)$ in the input proof tree;
    if $q_j(u, v) \in I(T, CH_\Gamma)$ **then**
        return $q_i(ax, by) \leftarrow q_j(x, y)$;
    else
        let P be the proof tree of $q_j(u, v)$ on Γ;
        return **contradiction_backtracing**(P).

next_clause :

Input: A clause $q_i(ax, by) \leftarrow q_j(x, y)$.
Output: A clause $q_i(ax, by) \leftarrow q_{j+1}(x, y)$.
Procedure:
  if $j = State$ **then**
    $State := State + 1$;
    let u, v be the strings such that $(q_i, (u, v)) \in CH_\Gamma$;
    $CH_\Gamma := CH_\Gamma \cup \{(q_{j+1}, (ua, vb))\}$;
  return $q_i(ax, by) \leftarrow q_{j+1}(x, y)$.

uncovered_atom :

Input: An atom α such that $\alpha \in I(T, CH_\Gamma)$ but $\alpha \notin M(\Gamma)$.
Output: An atom in $I(T, CH_\Gamma)$ which is not covered by any clause in Γ with respect to $I(T, CH_\Gamma)$.
Procedure:
  if there exists a clause $q_i(ax, by) \leftarrow q_j(x, y)$ such that
    $q_i(ax, by)\theta = \alpha$ and $q_j(x, y)\theta \in I(T, CH_\Gamma)$ for some substitution θ **then**
    return **uncovered_atom**($q_j(x, y)\theta$);
  else return α.

search_clause :

Input: An uncovered atom $q_i(u, v)$ which is returned by the procedure **uncovered_atom**.
Output: A new clause C whose head is unifiable with $q_i(u, v)$.
Procedure:
  if $u = v = \varepsilon$ **then**
    return $q_i(\varepsilon, \varepsilon)$;
  else let $u = au'$, $v = bv'$;
    if there exists $C \in \Gamma$ such that $head(C) = q_i(ax, by)$ **then**
      $\Gamma := \Gamma - \{C\}$;
      return **next_clause**(C);
    return $q_i(ax, by) \leftarrow q_0(x, y)$.

Here, we explain how the algorithm works. There are following two cases in which a hypothesis Γ should be modified.

1. The hypothesis is too strong, that is, $M(\Gamma)$ contains some negative fact.
2. The hypothesis is too weak, that is, $M(\Gamma)$ does not contain some positive fact.

In the case 1, there exists at least one clause in Γ which is not true in $I(T, CH_\Gamma)$. The algorithm find such a clause using the procedure **contradiction_backtracing**. The clause is removed from Γ. Then an alternate clause constructed by the procedure **next_clause** is added to the hypothesis.

In the case 2, there exists a ground atom $q_i(w_1, w_2) \in I(T, CH_\Gamma)$ which is not covered by any clause in Γ with respect to $I(T, CH_\Gamma)$. The procedure **uncovered_atom** finds out such an atom, and the procedure **search_clause** outputs a clause which covers the atom. If the uncovered atom is $q_i(\varepsilon, \varepsilon)$ then the procedure outputs $q_i(\varepsilon, \varepsilon) \leftarrow$. If the uncovered atom is $q_i(au, bv)$ such that there is no clause in Γ whose head is unifiable with $q_i(au, bv)$, then the procedure outputs the clause $q_i(ax, by) \leftarrow q_0(x, y)$. If the uncovered atom is $q_i(au, bv)$ and there exists a clause $C = q_i(ax, by) \leftarrow q_j(x, y)$ in Γ, then C is insufficient in $I(T, CH_\Gamma)$. Thus, C is removed from Γ and the procedure outputs **next_clause**(C).

In the algorithm, the predicate characterization is represented as a set of pairs of the form $(q, (u, v))$ where $q \in \Pi_\Gamma$ and $u, v \in \Sigma^*$. Let $CH_\Gamma = \{(q_0, (\varepsilon, \varepsilon)), (q_1, (u_1, v_1)), \ldots, (q_k, (u_k, v_k))\}$. Now we assume that the algorithm finds out a clause $q_i(ax, by) \leftarrow q_k(x, y)$ in Γ which is not complete in $I(T, CH_\Gamma)$. Then C is removed from Γ and the alternate clause $q_i(ax, by) \leftarrow q_{k+1}(x, y)$ is added to Γ. When the new predicate symbol q_{k+1} is introduced, the algorithm adds the pair $(q_{k+1}, (u_k a, v_k b))$ to CH_Γ.

3.3 Correctness of the algorithm

We prove that the learning algorithm is correct. First, we give a proposition concerned with the property of a predicate characterization.

Proposition 7 *Let Γ_i $(i = 1,2)$ be any DR-TEFS such that $\Pi_{\Gamma_1} \subseteq \Pi_{\Gamma_2}$. Suppose that $CH_{\Gamma_1}(q) = CH_{\Gamma_2}(q)$ for any $q \in \Pi_{\Gamma_1}$. Then, for any translation T and $q(w_1, w_2) \in \mathcal{B}(\Gamma_1)$, $q(w_1, w_2) \in I(T, CH_{\Gamma_1})$ if and only if $q(w_1, w_2) \in I(T, CH_{\Gamma_2})$.*

For the procedure **contradiction_backtracing**, we do not consider the atom of the form $q_i(\varepsilon, \varepsilon)$ as its input. Because, at any time on the learning process, there is no case in which $q_i(\varepsilon, \varepsilon)$ is in $M(\Gamma)$ but not in $I(T, CH_\Gamma)$. The ground atom $q_i(\varepsilon, \varepsilon)$ is in $M(\Gamma)$ if and only if there exists a clause $q_i(\varepsilon, \varepsilon) \leftarrow$ in Γ. The clause $q_i(\varepsilon, \varepsilon) \leftarrow$ is added to Γ after the procedure **search_clause** is called with the input $q_i(\varepsilon, \varepsilon)$. Let Γ' be the hypothesis for which the procedure call is occurred. Then, $q_i(\varepsilon, \varepsilon)$ is in $I(T, CH_{\Gamma'})$. By the Proposition 7, $q_i(\varepsilon, \varepsilon)$ is ensured to be in $I(T, CH_\Gamma)$ for any subsequent hypothesis Γ. Hence, there is no case in which $q_i(\varepsilon, \varepsilon)$ is in $M(\Gamma)$ but not in $I(T, CH_\Gamma)$.

Lemma 8 *Suppose that the procedure* **contradiction_backtracing** *is called with the proof tree of $\alpha \in M(\Gamma)$ such that $\alpha \notin I(T, CH_\Gamma)$. Then, the procedure returns a clause in Γ which is not true in $I(T, CH_\Gamma)$.*

Proof: Suppose that the procedure **contradiction_backtracing** given a proof tree of $q_i(au, bv)$ returns a clause $q_i(ax, by) \leftarrow q_j(x, y)$, where $a, b \in \Sigma$ and $u, v \in \Sigma^*$. Then, it is ensured that $q_i(au, bv) \notin I(T, CH_\Gamma)$ but $q_j(u, v) \in I(T, CH_\Gamma)$. Hence the clause is ensured to be false in the extended model $I(T, CH_\Gamma)$.

On the other hand, every input proof tree has the leaf $q_k(\varepsilon, \varepsilon)$. From discussion above, it is ensured that $q_k(\varepsilon, \varepsilon) \in I(T, CH_\Gamma)$. Since the input proof tree of each recursive call clear the input condition, a clause which is false in $I(T, CH_\Gamma)$ must be found eventually. □

Lemma 9 *Suppose that the procedure* **uncovered_atom** *is called with an input $\alpha \in I(T, CH_\Gamma)$ such that $\alpha \notin M(\Gamma)$. Then, the procedure returns $\alpha' \in I(T, CH_\Gamma)$ such that α' is not covered by any clause in Γ with respect to $I(T, CH_\Gamma)$.*

Proof: Since the procedure examines if $q_j(x, y)\theta \in I(T, CH_\Gamma)$ before calling itself recursively, every input $q_j(x, y)\theta$ for its recursive call is ensured to be in $I(T, CH_\Gamma)$. On the other hand, if an input $q_j(x, y)\theta$ for its recursive call in $M(\Gamma)$, then the input has a proof tree on Γ. This implies that all ancestors of the input have also proof tree on Γ. This contradicts that $\alpha \notin M(\Gamma)$. Thus, every input for its recursive call is not in $M(\Gamma)$.

Since $q_i(\varepsilon, \varepsilon)$ is not unifiable with any $q_i(ax, by)$, the procedure called with an input of the form $q_i(\varepsilon, \varepsilon)$ returns the input directly. Since $q_i(\varepsilon, \varepsilon) \notin M(\Gamma)$, it holds that $q_i(\varepsilon, \varepsilon) \leftarrow \notin \Gamma$. For any clause in DR-TEFS, $q_i(\varepsilon, \varepsilon)$ is covered only by the clause $q_i(\varepsilon, \varepsilon) \leftarrow$. Thus, if the procedure is called with the input $q_i(\varepsilon, \varepsilon)$, then it immediately follows that $q_i(\varepsilon, \varepsilon)$ is not covered by any clause in Γ with respect to $I(T, CH_\Gamma)$.

For an input of the form $q_i(au, bv)$, if there is no clause whose head is unifiable with $q_i(au, bv)$, it is clear that $q_i(au, bv)$ is not covered by any clause in Γ. Since a clause of the form $q_i(ax, by) \leftarrow q_{j+1}(x, y)$ is introduced into a hypothesis after the clause $q_i(ax, by) \leftarrow q_j(x, y)$ is removed from the hypothesis, there is at most one clause whose head is unifiable with $q_i(au, bv)$. Thus, if $q_i(au, bv)$ is not covered such a clause, there is no other clause which can cover $q_i(au, bv)$ with respect to $I(T, CH_\Gamma)$. Thus, if the procedure makes an output, then the output is ensured to satisfy the claim of the lemma.

On the other hand, the size of each input $q_j(x, y)\theta$ for the recursive call is decreasing two at a time. Thus, even if in the worst case, the procedure will encounter an input of the form $q_i(\varepsilon, \varepsilon)$ and terminate. This completes the proof of the lemma. □

Next, we show the justification of the way of constructing a predicate characterization. We can restate Proposition 7 as follows.

Proposition 10 *Let Γ_i $(i = 1, 2)$ be any DR-TEFS such that $\Pi_{\Gamma_1} \subseteq \Pi_{\Gamma_2}$. Suppose that $CH_{\Gamma_1}(q) = CH_{\Gamma_2}(q)$ for any $q \in \Pi_{\Gamma_1}$. Then, for any translation T and $C \in \Gamma_1$, C is complete in $I(T, CH_{\Gamma_1})$ if and only if C is complete in $I(T, CH_{\Gamma_2})$.*

Lemma 11 *Let Γ be a DR-TEFS. Suppose that $CH_\Gamma(q_i) = (u, v)$ and $CH_\Gamma(q_j) = (ua, vb)$ for some $a, b \in \Sigma$. Then, for any translation T, the clause $q_i(ax, by) \leftarrow q_j(x, y)$ is complete in $I(T, CH_\Gamma)$.*

Proof: By the definition of the extended model, for any $u, v \in \Sigma^*$, it holds that

$$\begin{aligned} q_i(aw_1, bw_2) \in I(T, CH_\Gamma) &\iff (uaw_1, vbw_2) \in T \quad (\text{where } CH_\Gamma(q_i) = (u, v)) \\ &\iff q_j(w_1, w_2) \in I(T, CH_\Gamma) \quad (\text{from } CH_\Gamma(q_j) = (ua, vb)) \end{aligned}$$

Hence, from Proposition 4, $q_i(ax, by) \leftarrow q_j(x, y)$ is complete in $I(T, CH_\Gamma)$. □

Theorem 12 *The predicate characterization CH_Γ constructed by the algorithm is, at any time, consistent and $CH_\Gamma(q_0) = (\varepsilon, \varepsilon)$.*

Proof: It is clear that $CH_\Gamma(q_0) = (\varepsilon, \varepsilon)$.

From the way of constructing the predicate characterization, for any $q_j \in \Pi_\Gamma$ $(j \geq 1)$, there exists $q_i \in \Pi_\Gamma$ such that $CH_\Gamma(q_i) = (u, v)$ and $CH_\Gamma(q_j) = (ua, vb)$ for some $a, b \in \Sigma$. On the other hand, in defining $CH_\Gamma(q_j)$, the clause $q_i(ax, by) \leftarrow q_j(x, y)$ is added to Γ simultaneously. Since Lemma 11 ensures that the clause is complete in $I(T, CH_\Gamma)$, it is never removed from the hypothesis. Hence, for any $q_j \in \Pi_\Gamma$ $(j \geq 1)$, there exists clauses in Γ that are necessary for constructing a derivation tree as in the definition of the consistency of CH_Γ.

Since $CH_\Gamma(q_0) = (\varepsilon, \varepsilon)$, $q_0(\varepsilon, \varepsilon)$ itself gives a derivation tree as in the definition of the consistency of CH_Γ. This holds even if $\Gamma = \emptyset$. Thus, the theorem holds. □

Lemma 13 *Let Γ be any hypothesis and CH_Γ be the predicate characterization for Γ constructed by Algorithm 1. For any $q_i \in \Pi_\Gamma$, there exist $w_1, w_2 \in \Sigma^*$ such that $(uw_1, vw_2) \in T$ where $CH_\Gamma(q_i) = (u, v)$.*

Proof: First, we show that, for any clause $q_k(ax, by) \leftarrow q_j(x, y) \in \Gamma$, there exist $w_1, w_2 \in \Sigma^*$ such that $q_k(aw_1, bw_2) \in I(T, CH_\Gamma)$. The clause whose head is $q_k(ax, by)$ first appears in a hypothesis after executing the last **else** statement in the procedure call of **search_clause** on the input $q_k(aw_1, bw_2)$. Let Γ' be the algorithm's hypothesis at that time. Then, $q_k(aw_1, bw_2) \in I(T, CH_{\Gamma'})$ and $q_k(aw_1, bw_2) \notin M(\Gamma')$. By the Proposition 7, for any subsequent CH_Γ, it holds that $q_k(aw_1, bw_2) \in I(T, CH_\Gamma)$. Thus, for any clause $q_k(ax, by) \leftarrow q_j(x, y)$, there exist $w_1, w_2 \in \Sigma^*$ such that $q_k(aw_1, bw_2) \in I(T, CH_\Gamma)$.

On the other hand, for any $q_i \in \Pi_\Gamma$ $(i \geq 1)$, there exists a predicate symbol $q_k \in \Pi_\Gamma$ such that $CH_\Gamma(q_k) = (u, v)$ and $CH_\Gamma(q_i) = (ua, vb)$ for some $a, b \in \Sigma$. By the argument in the proof of Theorem 12, there exists the clause $q_k(ax, by) \leftarrow q_i(x, y) \in \Gamma$. By the above discussion, there exist $w_1, w_2 \in \Sigma^*$ such that $q_k(aw_1, bw_2) \in I(T, CH_\Gamma)$, that is, $(uaw_1, vbw_2) \in T$. Hence, for any $q_i \in \Pi_\Gamma$ $(i \geq 1)$, there exists $w_1, w_2 \in \Sigma^*$ such that $CH_\Gamma(q_i) = (ua, vb)$ and $(uaw_1, vbw_2) \in T$.

For the predicate q_0, since T is not empty and $CH_\Gamma(q_0) = (\varepsilon, \varepsilon)$, there exist $w_1, w_2 \in \Sigma^*$ such that $(w_1, w_2) \in T$. □

Lemma 14 *Let $\hat{\Gamma}$ be a DR-TEFS with the minimum number of predicate symbols such that $T = T(\hat{\Gamma}, q_0)$. Let Γ be an arbitrary hypothesis constructed by the algorithm. Then it follows that $|\Pi_\Gamma| \leq |\Pi_{\hat{\Gamma}}|$.*

Proof: Let CH_Γ be the predicate characterization for Γ constructed by the algorithm. From Lemma 13, for any $q_i \in \Gamma$ such that $CH_\Gamma(q_i) = (u_i, v_i)$, there exist $w_1, w_2 \in \Sigma^*$ such that

$(u_i w_1, v_i w_2) \in T$. Since $T = T(\hat{\Gamma}, q_0)$, there uniquely exists a proof tree of $q_0(u_i w_1, v_i w_2)$ on $\hat{\Gamma}$. Hence, for any $q_i \in \Pi_\Gamma$, there uniquely exists a predicate symbol $\hat{q}_i \in \Pi_{\hat{\Gamma}}$ such that $\hat{q}_i(\varepsilon, \varepsilon)$ appears in the derivation tree of $q_0(u_i, v_i)$ on $\hat{\Gamma}$. For such $\hat{q}_i$, it holds that, for any $w_1, w_2 \in \Sigma^*$,

$$\hat{q}_i(w_1, w_2) \in M(\hat{\Gamma}) \iff q_0(u_i w_1, v_i w_2) \in M(\hat{\Gamma}) \quad (1)$$

Now, we consider the mapping τ from Π_Γ to $\Pi_{\hat{\Gamma}}$ such that $\tau(q_i) = \hat{q}_i$. For the proof of the lemma, it is sufficient to show that τ is injective.

Suppose that $\tau(q_i) = \tau(q_j)$ for some $i < j$ and $CH_\Gamma(q_i) = (u_i, v_i)$ and $CH_\Gamma(q_j) = (u_j, v_j)$. Then, for any $w_1, w_2 \in \Sigma^*$, it holds that

$$\begin{aligned} q_0(u_i w_1, v_i w_2) \in M(\hat{\Gamma}) &\iff \hat{q}_i(w_1, w_2) \in M(\hat{\Gamma}) \quad \text{(from 1)} \\ &\iff \hat{q}_j(w_1, w_2) \in M(\hat{\Gamma}) \quad \text{(from the assumption)} \\ &\iff q_0(u_j w_1, v_j w_2) \in M(\hat{\Gamma}) \quad \text{(from 1)}. \end{aligned}$$

Hence, we obtain the following relation.

$$(u_i w_1, v_i w_2) \in T \iff (u_j w_1, v_j w_2) \in T.$$

Since $0 \le i < j$, there exists $q_k \in \Pi_\Gamma$ $(k < j)$ such that $CH_\Gamma(q_k) = (u_k, v_k)$ and $CH_\Gamma(q_j) = (u_k a, v_k b)$ for some $a, b \in \Sigma$. Hence, for any $w_1, w_2 \in \Sigma^*$, the following relation holds.

$$(u_i w_1, v_i w_2) \in T \iff (u_j w_1, v_j w_2) \in T \iff (u_k a w_1, v_k b w_2) \in T.$$

As a result, it holds that

$$q_i(w_1, w_2) \in I(T, CH_\Gamma) \iff q_k(a w_1, b w_2) \in I(T, CH_\Gamma).$$

Hence, it follows from Proposition 4 that the clause $C = q_k(ax, by) \leftarrow q_i(x, y)$ is complete in $I(T, CH_\Gamma)$. Since $i < j$, C is generated by the procedure **next_clause** and added to the hypothesis before the clause $q_k(ax, by) \leftarrow q_j(x, y)$. From Proposition 10, C is complete in any extended model subsequently. Thus, C is never removed from subsequent hypothesis. This contradicts that $CH_\Gamma(q_j) = (u_k a, v_k b)$. □

Theorem 15 *For any DR-translation T, Algorithm 1 outputs a DR-TEFS Γ such that $T(\Gamma, q_0) = T$.*

Proof: It is clear that the procedures **next_clause** and **search_clause** terminated finitely and return the desired output. It follows from Lemma 8 and Lemma 9 that the procedures **contradiction_backtracing** and **uncovered_atom** terminate finitely and return the desired output. Hence, each computation in the bodies of two **if** statements terminates finitely and one of following operations is executed.

1. A clause of the form $q_i(\varepsilon, \varepsilon) \leftarrow$ is added to Γ.
2. A clause of the form $q_i(ax, by) \leftarrow q_0(x, y)$ is added to Γ.
3. A clause $q_i(ax, by) \leftarrow q_j(x, y)$ in Γ is replaced by $q_i(ax, by) \leftarrow q_{j+1}(x, y)$.

Both operations 1 and 2 are executed at most once for each i and $a, b \in \Sigma$. Hence, if only finitely many clauses are generated, then it is necessary that the two **if** statements are entered at most finitely many times in total. This means that the minimally adequate teacher replies "*yes*" in finite time.

On the other hand, by Proposition 10 and Lemma 11, once a predicate symbol is introduced into a hypothesis, the symbol never disappears from the subsequent hypothesis. Hence, by Lemma 14, only finitely many predicate symbols are generated.

This completes the proof of the theorem. □

3.4 Time complexity of the algorithm

We assume that the given teacher answers each membership query and equivalence query immediately. Then, we can obtain the following result. Let $\hat{\Gamma}$ be a DR-TEFS with the minimum number of predicate symbols such that $T = T(\hat{\Gamma}, q_0)$ and $|\Pi_{\hat{\Gamma}}| = n$.

Theorem 16 *For any DR-translation T, at any point during the run, the time used by Algorithm 1 to that point is bounded by some polynomial in n and the length of the longest counterexample returned by any equivalence query seen to that point.*

Proof: Let Γ be any hypothesis and CH_Γ be the predicate characterization for Γ constructed by the algorithm. For notational convenience, we denote $|\Sigma|$ by k and the maximum length of w_1 for any counterexample (w_1, w_2) given so far by m. Then, from Lemma 14, it follows that $|\Gamma| \leq n(k^2+1)$ and $|CH_\Gamma| \leq n$.

The both procedure **next_clause** and **search_clause** just make a simple search in CH_Γ and Γ respectively. Thus the time required in each procedure call is bounded by a linear in $n(k^2+1)$.

From the structure of a proof tree on a DR-TEFS, the input proof tree of some negative fact (w_1, w_2) for the procedure **contradiction_backtracing** can be treated as a sequence with length at most $|w_1|$. Since the procedure just traces the sequence, then the procedure terminates and finds a false clause in time linear in m.

For an input ground atom $q_i(w_1, w_2)$, the procedure **uncovered_atom** searches a clause in Γ whose head is unifiable with $q_i(w_1, w_2)$. If such a clause found then it calls itself recursively with input $q_j(u, v)$ such that $w_1 = au$ and $w_2 = bv$ for some $a, b \in \Sigma$. Otherwise, it returns the input directly. Since the main operation executed in the procedure is to search the clause, the time required in the procedure call is bounded by a linear in $|w_1| \times |\Gamma| \leq mn(k^2+1)$.

Each examination of whether given counterexample is positive or negative can be done immediately, because for a counterexample α, if $\alpha \in T$ then α is positive else α is negative. Hence, the time required in executing the body of each **if** statement in Algorithm 1 is bounded by a linear in $mn(k^2+1)$. On the other hand, since the number of possible clauses of a DR-TEFS constructed at most n predicate symbols is k^2n^2+n, each body of **if** statement is entered at most k^2n^2+n times.

Consequently, the amount of the time required in each iteration of the outer **repeat** loop is at most $O((k^2n^2+n)(mn(k^2+1))) = O(k^4mn^3)$. □

4 Conclusion

We gave a polynomial time learning algorithm for DR-translations using equivalence and membership queries. There are many cases where unobserved sub-concepts are necessary for representing a target translation. In this paper, we assumed that the minimally adequate teacher can answer questions about the target translation, that is, the partial Herbrand model of a target DR-TEFS. In the algorithm, new predicate symbols which correspond to such sub-concepts are produced. This depends on the property of DR-TEFS's. If we consider learning of more complex classes of translations, the problem might become more difficult. To extend target classes is one of the future problems.

In this paper, we focus on a subclass of translations called DR-translations, but an EFS is so powerful as a language generator that we can define various classes of translations larger than it. Directly, we can consider two natural extensions of DR-TEFS's. One is a TEFS which defines a translation between strings with different length. The other is a TEFS which defines a translation over larger class of languages. For practical applications, it is important to develop learning algorithms for such richer classes of translations.

References

[1] A. Aho and J. Ullman. Properties of syntax directed translations. *J. Comput. System Sci.*, 3:319–334, 1969.

[2] A. Aho and J. Ullman. Syntax directed translation and the pushdown assembler. *J. Comput. System Sci.*, 3:37–56, 1969.

[3] D. Angluin. Learning regular sets from queries and counterexamples. *Infomation and Computation*, 75:87–106, 1987.

[4] S. Arikawa, T. Shinohara, and A. Yamamoto. Elementary formal systems as a unifying framework for language learning. In *Proc. COLT'89*, pages 312–327, 1989.

[5] S. Arikawa, T. Shinohara, and A. Yamamoto. Learning elementary formal systems. *Theoret. Comput. Sci.*, pages 97–113, 1992.

[6] R. B. Banerji. Learning theories in a subset of a polyadic logic. In *Proc. Computational Learning Theory '88*, pages 281–295, 1988.

[7] S. Ginsburg. *The mathematical theory of context-free languages.* McGraw-Hill, 1966.

[8] H. Ishizaka. Inductive inference of regular languages based on model inference. *International journal of Computer Mathematics*, 27:67–83, 1989.

[9] H. Ishizaka. Polynomial time learnability of simple deterministic languages. *Machine Learning*, 5(2):151–164, 1990.

[10] X. Ling. Inventing theoretical terms in inductive learning of functions - search and constructive methods. In Z. W. Ras, editor, *Methodologies for Intelligent Systems, 4*, pages 332–341. North-Holland, October 1989.

[11] S. Muggleton and W. Buntine. Machine invention of first-order predicates by inverting resolution. In *Proc. 5th International Conference on Machine Learning*, pages 339–352, 1988.

[12] E. Shapiro. Inductive inference of theories from facts. Research report 192, Yale Univ., 1981.

[13] T. Shinohara. Rich classes inferable from positive data: length-bounded elementary formal system. *Information and computation*, 108:175–186, 1994.

[14] R. Smullyan. Theory of formal systems. *Princeton Univ. Press*, 1961.

[15] N. Sugimoto. Properties of elementary formal systems as translation grammars. Technical report rifis-tr-cs 107, Kyushu Univ., 1995.

[16] A. Yamamoto. Procedural semantics and negative information of elementary formal system. *Journal of Logic Programming*, 13:89–97, 1992.