Learning Bounded Unions of Noetherian Closed Set Systems Via Characteristic Sets

Yuuichi Kameda[1], Hiroo Tokunaga[1], and Akihiro Yamamoto[2]

[1] Department of Mathematics and Information Sciences
Graduate School of Science and Engineering,
Tokyo Metropolitan University
1-1 Minami-Ohsawa, Hachioji-shi 192-0397 Japan
tokunaga@tmu.ac.jp, kameda-yuuiti@ed.tmu.ac.jp
[2] Graduate School of Informatics, Kyoto University
Yoshida Honmachi, Sakyo-ku, Kyoto 606-850 Japan
akihiro@i.kyoto-u.ac.jp

Abstract. In this paper, we study a learning procedure from positive data for bounded unions of certain class of languages. Our key tools are the notion of characteristic sets and hypergraphs. We generate hypergraphs from given positive data and exploit them in order to find characteristic sets.

1 Introduction

In this paper, we study a learning procedure from positive data for a certain class of languages. In the following, "learning" always means "learning from positive data." The class of languages we consider is so called a *closed set system* (see §2 for its definition). In [4], we studied inferability of a closed set system in order to understand relations between the class of ideals of the polynomial ring and inferability from positive data. The polynomial ring is a fundamental object in algebra, and Hilbert's basis theorem about the finite generation of its ideals has been historically important. In [5], Hayashi pointed out that if we consider ideals of the polynomial ring as formal languages, the statement of Hilbert's basis theorem can be understood as inferability from positive data. In fact, Stephan and Ventsov [9] showed that a finite basis of any ideal of a commutative ring is regarded as a finite tell-tale. In [4], we introduced a notion of a *Noetherian closed set system* and proved that a closed set system $\mathcal{L}$ has finite elasticity if and only if it is Noetherian. Hence by the result of Wright et al. (Theorem 2.4 in §2), the class of bounded union $\cup^{\leq k}\mathcal{L}$ also has finite elasticity. From the proof of Theorem 2.4, however, we do not know how its learning procedure looks like. On the other hand, Kobayashi introduced the notion of a *characteristic set* in [7] and proved that (i) if the class $\mathcal{L}$ has finite elasticity, then every language in $\mathcal{L}$ has a characteristic set and (ii) $\mathcal{L}$ is inferable from positive data if every language in $\mathcal{L}$ has a characteristic set. Our goal of this note is to give a learning

A. Clark, F. Coste, and L. Miclet (Eds.): ICGI 2008, LNAI 5278, pp. 98–110, 2008.

procedure for bounded unions of certain class of Noetherian closed set systems by using characteristic sets.

The contents of this paper is as follows. In §2, we summarize some facts on inferablity from positive data and closed set systems. In §3, we give a learning procedure of bounded unions of a Noetherian closed set system under certain settings as follows:

(i) Given a closed set $L \in \mathcal{L}$ and its characteristic set F, there exists an algorithm to compute a characteristic set of L in $\cup^{\leq k}\mathcal{L}$ from F.
(ii) $\cup^{\leq k}\mathcal{L}$ is compact (See §2 for its definition).

Our procedure is given by generating a certain hypergraph, which is main feature of our result. In §§4 and 5, we apply our procedure to bounded unions of the class of ideals of the polynomial ring and tree pattern languages, respectively.

2 Preliminaries

2.1 Inferability from Positive Data

In this article, a *language* L is a subset of some countable set U such that L is expressed $L(G)$ by some finite expression G. We call this finite expression a *hypothesis*. A set of all hypotheses $\mathcal{H}$ is called a *hypothesis space*. Let $\mathcal{L}$ be the set of all languages $\{L(G) \mid G \in \mathcal{H}\}$. We assume that $\mathcal{L}$ is *uniformly recursive*, that is, there is a recursive function $f(w, G)$ such that $f(w, G) = 1$ iff $w \in L(G)$ for every $w \in U$ and $G \in \mathcal{H}$.

A *positive data* (or *positive presentation*) of $L \in \mathcal{L}$ is an infinite sequence $\sigma : s_1, s_2, \ldots$ of elements of L such that $L = \{s_1, s_2, \ldots\}$. An *inference algorithm* M is that:

- M receives incrementally an elements of a positive data σ of a language,
- M outputs a hypothesis $G_n \in \mathcal{H}$ when M receives n-th element of σ.

$\mathcal{L}$ is *inferable in the limit from positive data* if there exists an inference algorithm M satisfies that for all $L \in \mathcal{L}$ and an arbitrary positive data of L, the output sequence of M converges to a hypothesis G such that $L(G) = L$.

A *finite tell-tale* of $L \in \mathcal{L}$ is a finite subset S of L such that L is a minimal in the class $\{L' \in \mathcal{L} \mid S \subset L'\}$ with respect to set inclusion. If L is minimum, S is called a *characteristic set* of L. Note that the idea of characteristic set is essentially the same as that of *test set* in [6].

Theorem 2.1. ([1]) *$\mathcal{L}$ is inferable in the limit from positive data if and only if there exists a procedure to enumerate elements of a finite tell-tale of every $L \in \mathcal{L}$.*

Theorem 2.2. ([7]) *If every $L \in \mathcal{L}$ has a characteristic set, then $\mathcal{L}$ is inferable from positive data.*

We say that (i) $\mathcal{L}$ has *finite thickness* if the set $\{L \in \mathcal{L} \mid w \in L\}$ is finite for any $w \in U$ and (ii) $\mathcal{L}$ has *infinite elasticity* if there exists an infinite sequence $w_0, w_1, \dots$ of elements of U and infinite sequence $L_1, L_2, \dots$ of languages such that $\{w_0, \dots, w_{n-1}\} \subset L_n$ but $w_n \notin L_n$. We say that $\mathcal{L}$ has *finite elasticity* if it does not have infinite elasticity.

Theorem 2.3. ([7],[8]) (1) *If $\mathcal{L}$ has finite elasticity, then every L in $\mathcal{L}$ has a characteristic set.*
(2) *If $\mathcal{L}$ has finite thickness, then $\mathcal{L}$ has finite elasticity.*

We define a class of the union of languages as follows:

$$\mathcal{L} \cup \mathcal{L}' = \{L_1 \cup L_2 \mid L_1 \in \mathcal{L}, L_2 \in \mathcal{L}'\},$$
$$\cup^{\leq k}\mathcal{L} = \{L_1 \cup \ldots \cup L_m \mid m \leq k, L_i \in \mathcal{L}\ (i = 1, \dots, m)\}.$$

It is known that

Theorem 2.4. ([12]) *If $\mathcal{L}$ and $\mathcal{L}'$ have finite elasticity, then $\mathcal{L} \cup \mathcal{L}'$ has finite elasticity.*

It immediately follows that, if $\mathcal{L}$ has finite elasticity, then $\cup^{\leq k}\mathcal{L}$ also has. Therefore, by Theorems 2.2 and 2.3, we have:

Corollary 2.1. *If $\mathcal{L}$ has finite elasticity, then $\cup^{\leq k}\mathcal{L}$ is inferable from positive data.*

Definition 2.1. *$\cup^{\leq k}\mathcal{L}$ is said to be compact if it satisfies the following condition:*
For each $m \leq k$ and $L, L_i \in \mathcal{L}$ $(i = 1, \dots, m)$, if $L \subset L_1 \cup \ldots \cup L_m$, then there exists i_0 such that $L \subset L_{i_0}$.

2.2 Closed Set System

Let 2^U be the power set of U. A mapping $C : 2^U \to 2^U$ is called a *closure operator* if C satisfies:
(CO1) $X \subset C(X)$,
(CO2) $C(C(X)) = C(X)$, and
(CO3) $X \subset Y \Rightarrow C(X) \subset C(Y)$,
where X and Y are arbitrary subsets of U. A set $X \subset U$ is called *closed* if $X = C(X)$. A *closed set system* $\mathcal{C}$ is the class of all closed sets of a closure operator.

Remark 2.1. In a closed set system, the intersection of arbitrary number of closed sets is closed, but the union of closed sets is not necessarily closed.

In the following, we regard $\mathcal{C}$ as a class of languages and assume that it is recursive. If a closed set $X \in \mathcal{C}$ is represented $X = C(Y)$ for some finite set $Y \subset U$, X is called a *finitely generated closed set.*

Lemma 2.1. ([4]) *Let $X = C(Y)$ be a closed set. The followings are equivalent:*
1. Y is finite,
2. Y is a finite tell-tale of X, and
3. Y is a characteristic set of X.

An immediate consequence of Lemma 2.1 and Theorem 2.1 is as follows:

Corollary 2.2. *$\mathcal{C}$ is inferable from positive data if and only if every closed set is finitely generated.*

A closed set system $\mathcal{C}$ is *Noetherian* if it contains no infinite strictly ascending chain of closed sets. This condition is equivalent to finite elasticity [4, Theorem 7]. Hence it follows that:

Corollary 2.3. *A Noetherian closed set system is inferable from positive data.*

3 Main Result

As for inferability of closed set systems from positive data, we refer to [4], and use result there freely.

Let $\mathcal{L}$ be a Noetherian closed set systems over some set U and C denote its closure operator. By [4, Theorem 7], $\mathcal{L}$ has finite elasticity and it implies that the class $\cup^{\leq k}\mathcal{L}$ also has finite elasticity. In particular, by [4, §3], any element L of $\mathcal{L}$ is of the form $L = C(F)$ for a finite subset F of U. In this section, we consider learning procedure for $\cup^{\leq k}\mathcal{L}$.

Remark 3.1. For an element $L_1 \cup \ldots \cup L_m \in \cup^{\leq k}\mathcal{L}$, we assume that $L_i \not\subseteq L_j$ for any $i, j (i \neq j)$.

Let F be a finite subset of U. F is a characteristic set of $C(F)$ in $\mathcal{L}$. Since $C(F)$ is also a member of $\cup^{\leq k}\mathcal{L}$, $C(F)$ has a characteristic set in $\cup^{\leq k}\mathcal{L}$, and we denote it by $\chi(C(F), \cup^{\leq k}\mathcal{L})$ (we may assume that $F \subseteq \chi(C(F), \cup^{\leq k}\mathcal{L})$). Throughout this section, we assume the following:

($*$) There exists an algorithm to compute $\chi(C(F), \cup^{\leq k}\mathcal{L})$ from F.

In §§4 and 5, we give examples of Noetherian closed set systems satisfying ($*$).

Let $L_1 \cup \ldots \cup L_m \in \cup^{\leq k}\mathcal{L}$ and let $\sigma : f_1, f_2, \ldots, f_n, \ldots$ be a positive data of $L_1 \cup \ldots \cup L_m$. We inductively define a hypergraph denoted by $\mathcal{G}_n$ having the set of vertices $V(\mathcal{G}_n) = \{f_1, \ldots, f_n\}$ as follows:

Inductive definition of $\mathcal{G}_n$
Let $V(\bullet)$ and $HE(\bullet)$ denote the set of vertices and hyperedges of a hypergraph $\bullet$, respectively.

For $n = 1$, we put

$$V(\mathcal{G}_1) = \{f_1\}, \quad HE(\mathcal{G}_1) = \{\{f_1\}\}.$$

Suppose that $\mathcal{G}_n$ is already given and f_{n+1} is presented. We construct $\mathcal{G}_{n+1}$ in the following way:

Procedure 1: Construction of $\mathcal{G}_{n+1}$ from $\mathcal{G}_n$;
Input: f_{n+1} and $\mathcal{G}_n$;
Output: a hypergraph $\mathcal{G}_{n+1}$;
begin
1. put $V = V(\mathcal{G}_n) \cup \{f_{n+1}\}$ and $HE = HE(\mathcal{G}_n)$;
2. **for** each subset $F \subset V$ such that $f_{n+1} \in F$ and $\sharp(F) \geq 2$ **do begin**
3. let $E = \chi(C(F), \cup^{\leq k}\mathcal{L})$;
4. **if** $E \subset V$ **then begin**
5. **for** each element $\mathcal{E}$ of HE **do**
6. **if** $\mathcal{E} \subset E$ **then** remove $\mathcal{E}$ from HE;
7. add E to HE;
8. **end**;
9. **end**;
10. **if** there is no $\mathcal{E} \in HE$ such that $f_{n+1} \in \mathcal{E}$ **then** add $\{f_{n+1}\}$ to HE;
11. **return** $\mathcal{G}_{n+1} = (V, HE)$;
end.

Note that $\{f_{n+1}\}$ is a characteristic set of $C(\{f_{n+1}\})$ in $\cup^{\leq k}\mathcal{L}$.
We are now in a position to give our learning procedure:

Procedure 2: Learning $\cup^{\leq k}\mathcal{L}$;
Input: a positive presentation $\sigma : f_1, f_2, \ldots, f_n, \ldots$ for $L_1 \cup \ldots \cup L_m$;
Output: a sequence of at most k-tuples of characteristic sets
$(\chi_1^{(1)}, \ldots, \chi_{m_1}^{(1)}),\ (\chi_1^{(2)}, \ldots, \chi_{m_2}^{(2)}), \ldots$;
begin
1. $S = \emptyset$; /*Possible candidates for characteristic sets*/
2. Put $n = 1$;
3. **repeat**
4. construct the hypergraph $\mathcal{G}_n$ for $f_1, f_2, \ldots, f_n$;
5. put $S = HE(\mathcal{G}_n)$;
6. choose at most k maximal elements from S with respect to the order as below;
7. output (at most) k-tuple in 6;
8. add 1 to n;
9. **forever**;
end.

We define an ordering on S as follows:

$\chi_1 < \chi_2 \Leftrightarrow C(\chi_1) \subsetneq C(\chi_2)$
ELSE $C(\chi_1) = C(\chi_2)$ and $\chi_1 \prec \chi_2$ under a certain suitable ordering $\prec$.

The ordering $\prec$ does not affect the validity of **Procedure 2**, so we can adopt a convenient ordering (for example, the order of appearance in S.)

Remark 3.2. Note that $C(\chi_i^{(n)}) \not\subseteq C(\chi_j^{(n)})$ for any i, j $(i \neq j)$.

Now our theorem is the following:

Theorem 3.1. *Suppose that $\cup^{\leq k}\mathcal{L}$ is compact. $\cup^{\leq k}\mathcal{L}$ is identifiable in the limit from positive data via **Procedure 2**.*

We need some lemmas to prove Theorem 3.1.

Lemma 3.1. *Let $\mathcal{E}$ be an arbitrary hyperedge of $\mathcal{G}_n$. Then $C(\mathcal{E}) \subset L_1 \cup \ldots \cup L_m$. Moreover, if $\mathcal{L}$ is compact, then there exists L_i such that $C(\mathcal{E}) \subseteq L_i$.*

Proof. By our construction of $\mathcal{G}_n$, there exists $F \subseteq V(\mathcal{G}_n)$ such that $\mathcal{E} = \chi(C(F), \cup^{\leq k}\mathcal{L})$. Since $\mathcal{E} \subset C(F)$, $C(\mathcal{E}) \subseteq C(C(F)) = C(F)$. On the other hand, $\mathcal{E}$ is also a characteristic set of $C(F)$ in $\mathcal{L}$, $C(\mathcal{E}) \supseteq C(F)$, i.e., $C(\mathcal{E}) = C(F)$. Moreover, since $\mathcal{E}$ is a characteristic set of $C(F)$ in $\mathcal{L}^{\leq k}$, $C(F) \subseteq L_1 \cup \ldots \cup L_m$. The second statement is immediate from the definition of compactness.

Lemma 3.2. *Suppose that $\cup^{\leq k}\mathcal{L}$ is compact.* (1) *Let $L_1, \ldots, L_m$ be distinct members of $\mathcal{L}$. If $L_i \not\subset L_j$ for all $i, j (i \neq j)$, then $L_i \not\subset \cup_{j=1, j\neq i}^{m} L_j$.* (2) *Let $M \in \mathcal{L}$ and let $L_1, \ldots, L_m$ be as above. If $M \subseteq L_1 \cup \ldots \cup L_m$ and $L_i \subseteq M$ for some i, then $L_i = M$.*

Proof. (1) If $L_i \subset \cup_{j=1, j\neq i}^{m} L_j$, then $L_i \subseteq L_{j_0}$ for some j_0 by compactness. This contradicts to our assumption. (2) By compactness, there exists L_{j_0} such that $M \subseteq L_{j_0}$. Hence $L_i \subseteq M \subseteq L_{j_0}$. By our assumption, $L_i = L_{j_0}$.

Proof of Theorem 3.1. Let $L_1 \cup \ldots \cup L_m$ be an arbitrary element in $\cup^{\leq k}\mathcal{L}$. Suppose that $L_i = C(F_i)$, where F_i is a finite subset of L_i. Since F_i can be considered as $\chi(L_i, \mathcal{L})$, at a certain finite step N_0, all elements of F_i are presented. Therefore, at a certain step N after the step N_0, one can assume that all elements of $\chi(C(F_i), \cup^{\leq k}\mathcal{L})\ for\ i = 1, \ldots, m$ are presented. Let $\mathcal{E}_{1,N}, \ldots, \mathcal{E}_{m_N,N}$ be hyperedges of $\mathcal{G}_N$ as in Output of Procedure 2. By construction, each $\mathcal{E}_{j,N}$ is a characteristic set of closed set $C(\mathcal{E}_{j,N})$ contained in $L_1 \cup \ldots \cup L_m$. Note that $C(\mathcal{E}_{i,N}) \not\subseteq C(\mathcal{E}_{j,N})$ for any $i, j (i \neq j)$ by Remark 3.2.

Claim. For each $\chi(C(F_i), \cup^{\leq k}\mathcal{L})$, there exists a unique $\mathcal{E}_{j_i,N}$ of $HE(\mathcal{G}_N)$ with $C(F_i) = C(\mathcal{E}_{j_i,N})$.

By our construction of $\mathcal{G}_N$, $\chi(C(F_i), \cup^{\leq k}\mathcal{L})$ is either added as a hyperedge or contained in a hyperedge added at a certain step. Hence there exists a hyperedge $\mathcal{E}_i$ of $\mathcal{G}_N$ such that $\chi(C(F_i), \cup^{\leq k}\mathcal{L}) \subseteq \mathcal{E}_i$ for each i. Since $C(F_i) = C(\chi(C(F_i), \cup^{\leq k}\mathcal{L})) \subseteq C(\mathcal{E}_i)$, $C(\mathcal{E}_i) = C(F_i)$ by Lemmas 3.1 and 3.2(2). As $L_i = C(\mathcal{E}_i)$ is a maximal element of $\mathcal{L}$ contained in $L_1 \cup \ldots \cup L_m$, there exists a hyperedge $\mathcal{E}_{j,N}$ such that $L_i = C(F_i) = C(\mathcal{E}_{j,N})$ (Note that $\mathcal{E}_{j,N}$ is not necessarily equal to $\mathcal{E}_i$). Suppose that there exist two distinct $\mathcal{E}_{j_1,N}$ and $\mathcal{E}_{j_2,N}$ such that $\chi(C(F_i), \cup^{\leq k}\mathcal{L}) \subseteq \mathcal{E}_{j_l}$ $(l = 1, 2)$. Then $C(F_i) = C(\chi(C(F_i), \cup^{\leq k}\mathcal{L})) \subseteq C(\mathcal{E}_{j_l})$ $(l = 1, 2)$. By Lemma 3.2, $C(F_i) = C(\mathcal{E}_{j_1,N}) = C(\mathcal{E}_{j_2,N})$, but this contradicts to our assumption.

We finally show that $m_N = m$. By Claim, $m_N \geq m$. If $m_N > m$, then there exists $\mathcal{E}_{j_0,N}$ such that (i) $C(\mathcal{E}_{j_0,N}) \neq L_i$ for $i = 1, \ldots, m$ and (ii) $C(\mathcal{E}_{j_0,N}) \subset L_1 \cup \ldots \cup L_m$. But these condition implies that $C(\mathcal{E}_{j_0,N}) \subset C(\mathcal{E}_{j_i,N})$ for some j_i. This contradicts to our choice of $\mathcal{E}_{i,N}$ $(i = 1, \ldots, m_N)$.

Remark 3.3. Note that the hypotheses in our algorithm are not necessarily consistent. However, one can modify them into consistent ones without difficulty.

4 Learning Bounded Set Unions of Polynomial Ideals

We denote the set of polynomials of n variables with $\mathbb{Q}$-coefficients by $\mathbb{Q}[x_1, \ldots, x_n]$. A subset I of $\mathbb{Q}[x_1, \ldots, x_n]$ is called an *ideal* if it satisfies the following:

- For each $f, g \in I$, $f \pm g \in I$.
- For each $f \in I$ and $h \in \mathbb{Q}[x_1, \ldots, x_n]$, $hf \in I$.

We denote the set of all ideals by $\mathcal{I}$. For a finite subset $F = \{f_1, \ldots, f_r\} \subset \mathbb{Q}[x_1, \ldots, x_n]$, we define the ideal generated by $f_1, \ldots, f_r$, which is denoted by $\langle f_1, \ldots, f_r \rangle$ or $\langle F \rangle$, as follows:

$$\langle F \rangle := \Big\{ \sum_{i=1}^{r} h_i f_i \mid \ h_i \in \mathbb{Q}[x_1, \ldots, x_n] \Big\}.$$

Note that the correspondence $F \mapsto \langle F \rangle$ defines a closure operator on $\mathbb{Q}[x_1, \ldots, x_n]$. By Hilbert's basis theorem for polynomial ideals, we have the following: for each $I \in \mathcal{I}$, there exists a finite set F such that $I = \langle F \rangle$. An interpretation of this statement from machine learning view point is that "$\mathcal{I}$ has a finite elasticity." Hence, $\mathcal{I}$ is a Noetherian closed set system with the closure operator $F \mapsto \langle F \rangle$. Furthermore, the existence of a reduced Groebner basis for given I in theory of Groebner basis says that one can take the set of reduced Groebner bases as a hypothesis space of $\mathcal{I}$.

The following lemma is a special case of [10, Theorem 9]. This lemma implies that $\cup^{\leq k}\mathcal{I}$ satisfies the condition $(*)$.

Lemma 4.1. *Let $I \in \mathcal{I}$. A characteristic set $\chi(I, \cup^{\leq k}\mathcal{I})$ can be constructed if the reduced Groebner basis $G = \{g_1, \ldots, g_r\}$ of I is given.*

Remark 4.1. For instance we have an example of $\chi(I, \cup^{\leq k}\mathcal{I})$ as follows:

$$h_i = g_1 + c_i g_2 + \ldots + c_i^{r-1} g_r \ (i = 1, \ldots, M)$$

where $M = k(r-1)+1$ and c_i's are distinct elements of $\mathbb{Q}$. Note that no h_i will vanish since $\{g_1, \ldots, g_r\}$ is the reduced Groebner basis.

Remark 4.2. This lemma also implies that $\cup^{\leq k}\mathcal{I}$ is compact: suppose that $I = \langle g_1, \ldots, g_r \rangle$ is contained in $I_1 \cup \ldots \cup I_m$. Let $M = m(r-1)+1$ and take $h_1, \ldots, h_M$ as above. By the pigeon-hole principle, there exists some j such that I_j includes at least r of h_i's. This means $I \subset I_j$.

According to above arguments, we have:

Theorem 4.1. *$\cup^{\leq k}\mathcal{I}$ is identifiable in the limit from positive data via **Procedure 2**.*

Example 4.1. Let us consider learning $\langle x^2, y^3 \rangle \cup \langle x^3, y^2 \rangle \in \cup^{\leq 2}\mathcal{I}$. Let a positive presentation σ be $x^2, y^3, y^2, x^2+y^3, x^3, x^3+y^2, \ldots$. By the argument of §3 of [10], we can take a characteristic set $\chi(\langle f, g \rangle, \cup^{\leq 2}\mathcal{I}) = \{f, g, f+g\}$ for distinct polynomials f and g. The hyperedges of hypergraphs constructed by **Procedure 1** are as follows:

$$
\begin{aligned}
HE_1 &= \{\{x^2\}\}, \\
HE_2 &= \{\{x^2\}, \{y^3\}\}, \\
HE_3 &= \{\{x^2\}, \{y^3\}, \{y^2\}\}, \\
HE_4 &= \{\{x^2, y^3, x^2+y^3\}, \{y^2\}\}, \\
HE_5 &= \{\{x^2, y^3, x^2+y^3\}, \{y^2\}, \{x^3\}\}, \\
HE_6 &= \{\{x^2, y^3, x^2+y^3\}, \{y^2, x^3, x^3+y^2\}\}.
\end{aligned}
$$

Hence **Procedure 2** learns $\langle x^2, y^3 \rangle \cup \langle x^3, y^2 \rangle$ when $n = 6$.

5 Learning Bounded Unions of Tree Pattern Languages

In [3], Arimura et al. studied learnability of bounded union of tree pattern languages. However, they did not seem to use characteristic sets explicitly. We here give a procedure learning bounded unions of tree pattern languages by using our result in §3. Let Σ be a finite set and V be a countable set disjoint from Σ. The elements of Σ and V are called *symbols* and *variables*, respectively. We assume that there is a mapping *rank* that maps an element of Σ to a non-negative integers. We define the rank of elements of V to be zero. A *tree pattern* p over Σ is a tree satisfying following properties:

- p has the root.
- p is directed.
- p is ordered.
- Each node of p is labeled by elements of $\Sigma \cup V$.
- The number of children of each node is equal to the rank of the label of the node.

A *tree* over Σ is a tree pattern over Σ that has no nodes labeled by an element of V. $\mathcal{TP}_\Sigma$ and $\mathcal{T}_\Sigma$ denote the set of all tree patterns and all trees over Σ, respectively.

A *substitution* is a mapping θ from V to $\mathcal{TP}_\Sigma$. $p\theta$ denotes the tree pattern obtained from applying a substitution θ to p. We define a relation on $\mathcal{TP}_\Sigma$ as follows: $p \preceq q \Leftrightarrow$ there exists a substitution θ such that $p = q\theta$. We denote $p \equiv q$ if $p \preceq q$ and $q \preceq p$, and call that p and q are equivalent. Note that $p \equiv q$ if and only if $p = q\theta$ for some renaming θ of variables.

Lemma 5.1. (1) *If $p \preceq q$ and $q \preceq r$, then $p \preceq r$.*
(2) *Let $|p|$ be the number of nodes of p. If $p \preceq q$, then $|p| \geq |q|$.*
(3) *For any $p \in \mathcal{TP}_\Sigma$, there are finitely many $q \in \mathcal{TP}_\Sigma$ such that $p \preceq q$.*

Lemma 5.2. *For any subset $S \neq \emptyset$ of $\mathcal{TP}_\Sigma$, there exists an element $lca(S)$ of $\mathcal{TP}_\Sigma$ such that:*

(i) $p \preceq lca(S)$ for any $p \in S$,
(ii) if $p \preceq r$ for any $p \in S$, then $lca(S) \preceq r$.

$lca(S)$ is uniquely determined up to equivalence. It is called the least common anti-instance *of S. If S is finite, then $lca(S)$ can be computed in polynomial time [8].*

A *tree pattern language defined by p* is the set $L(p) = \{t \in \mathcal{T}_\Sigma \mid t \preceq p\}$. We denote the set of all tree pattern languages $\{L(p) \mid p \in \mathcal{TP}_\Sigma\}$ by $\mathcal{TPL}(\Sigma, V)$. We may omit (Σ, V) if it is clear from the context.

Lemma 5.3. (1) *$p \preceq q \Leftrightarrow L(p) \subset L(q)$ for any $p, q \in \mathcal{TP}_\Sigma$.*
(2) *$L(t) = \{t\}$ for any $t \in \mathcal{T}_\Sigma$.*
(3) *$lca(t_1, t_2) \preceq p$ for any $p \in \mathcal{TP}_\Sigma$ and $t_1, t_2 \in L(p)$.*

In general, the class $\mathcal{TPL}$ itself may be not a closed set system. Hence we introduce a closed set system $\mathcal{C}$ over $\mathcal{TPL}$.
For $S \subset \mathcal{TP}_\Sigma$, we define $C(S) = \{p \in \mathcal{TP}_\Sigma \mid p \preceq lca(S)\}$. Note that $C(S) = C(lca(S))$.

Lemma 5.4. *C is a closure operator on $\mathcal{TP}_\Sigma$.*

Proof. (CO1) Obvious by the definition of lca. (CO2) In general, $lca(C(S)) = lca(S)$ holds. Thus $C(C(S)) = C(lca(C(S))) = C(lca(S)) = C(S)$. (CO3) Suppose $S_1 \subset S_2 \subset \mathcal{TP}_\Sigma$. Clearly, $lca(S_1) \preceq lca(S_2)$. Lemma 5.1(1) implies $C(S_1) = C(lca(S_1)) \subset C(lca(S_2)) = C(S_2)$.

$\mathcal{C}$ denotes the closed set system defined by C. The following lemma indicates a fundamental relation between $\mathcal{TPL}$ and $\mathcal{C}$.

Lemma 5.5. *For every $p \in \mathcal{TP}_\Sigma$, $L(p) = C(p) \cap \mathcal{T}_\Sigma$.*

Lemma 5.6. (1) *$\mathcal{TPL}$ and $\mathcal{C}$ have finite elasticity.*
(2) *If $\sharp(\Sigma) > k$, $\cup^{\leq k}\mathcal{TPL}$ is compact.*
(3) *$\cup^{\leq k}\mathcal{C}$ is compact.*

Proof. (1) For any fixed $k \in \mathbb{N}$, the set $\{p \in \mathcal{TP}_\Sigma \mid |p| \le k\}$ is finite up to equivalence. This fact and Lemma 5.1(2) imply that, for any $p \in \mathcal{TP}_\Sigma$, there are finitely many $q \in \mathcal{TP}_\Sigma$ such that $p \preceq q$. This means that $\mathcal{TPL}$ and $\mathcal{C}$ have finite thickness. Therefore, they have finite elasticity by Theorem 2.3.
(2) See [2].
(3) Suppose that $C(p) \subset C(p_1) \cup \ldots \cup C(p_m)$ $(m \le k)$. Since $p \in C(p)$, there exists i_0 such that $p \in C(p_{i_0})$. Hence $C(p) \subset C(p_{i_0})$.

Let $\Sigma_0 = \{a \in \Sigma \mid rank(a) = 0\}$ and $\Sigma_+ = \{f \in \Sigma \mid rank(f) > 0\}$. In the following, we assume that neither Σ_0 nor Σ_+ is empty.

Lemma 5.7. (1) *For every $p \in \mathcal{TP}_\Sigma$, there exists a characteristic set $\chi(L(p), \mathcal{TPL})$ consisting of at most two elements.*
(2) *For every $S \subset \mathcal{TP}_\Sigma$, there exists a characteristic set $\chi(C(S), \mathcal{C})$ consisting of one element.*

Lemma 5.8. (1) *Suppose $\sharp(\Sigma_+) \ge k$. For every $p \in \mathcal{TP}_\Sigma$, there exists a characteristic set $\chi(L(p), \cup^{\le k}\mathcal{TPL})$ consisting of at most $k+1$ elements. (In fact, there exists a set $\{t_1, \ldots, t_{k+1}\} \subset \mathcal{T}_\Sigma$ such that $lca(t_i, t_j) = p$ for each $i \ne j$. See [11] for detail.)*
(2) *For every $S \subset \mathcal{TP}_\Sigma$, there exists a characteristic set $\chi(C(S), \cup^{\le k}\mathcal{C})$ consisting of one element.*

Lemma 5.8(2) makes algorithm learning $\cup^{\le k}\mathcal{C}$ much simpler.

Procedure 3: Learning $\cup^{\le k}\mathcal{C}$;
Input: a positive presentation $\sigma : q_1, q_2, \ldots, q_n, \ldots$ for
$C(p_1) \cup \ldots \cup C(p_m)$;
Output: a sequence of at most k-tuples of tree patterns
$(r_1^{(1)}, \ldots, r_{m_1}^{(1)}), (r_1^{(2)}, \ldots, r_{m_2}^{(2)}), \ldots$;
begin
1. $S = \emptyset$; /*The set to memorize a given sequence of $q_1, \ldots, q_n$*/
2. put $n = 1$;
3. **repeat**
4. add q_n to S;
5. choose at most k maximal elements from S with respect to $\preceq$ up to equivalence;
6. output (at most) k-tuple in 5;
7. add 1 to n;
8. **forever**

end.

We assume $\sharp(\Sigma_+) \ge k$ in order to make Lemma 5.8(1) holds. By using **Procedure 3**, $\cup^{\le k}\mathcal{TPL}$ is inferred as follows:

Procedure 4: Learning $\cup^{\leq k}\mathcal{TPL}$;
Input: a positive presentation $\sigma : t_1, t_2, \ldots, t_n, \ldots$ for
$L(p_1) \cup \ldots \cup L(p_m)$;
Output: a sequence of at most k-tuples of tree patterns
$(q_1^{(1)}, \ldots, q_{m_1}^{(1)}), (q_1^{(2)}, \ldots, q_{m_2}^{(2)}), \ldots$;
begin
1. generate "positive data" of $C(p_1) \cup \ldots \cup C(p_m)$ from σ;
2. run **Procedure 3** by "positive data" generated in 1;
3. output the output of 2;
end.
Generation of "positive data" ($\mathcal{GPD}$);
4. $S = \emptyset$; /*The set to memorize a given sequence of $t_1, \ldots, t_n$*/
5. put $n = 1$;
6. **repeat**
7. add t_n to S;
8. **output** t_n;
9. **for** each subset F of S with $t_n \in F$ and $\sharp(F) = k + 1$ **do**
10. **if** $lca(t_i, t_j) = lca(F)$ for all $t_i, t_j \in F$ $(i \neq j)$ **then**
11. **output** $lca(F)$;
12. add 1 to n;
13. **forever**;
end.

Theorem 5.1. *$\cup^{\leq k}\mathcal{TPL}$ is identifiable in the limit from positive data via* ***Procedure 4****.*

Proof. It suffices to show that $\mathcal{GPD}$ generates a positive data for $A = C(p_1) \cup \ldots \cup C(p_m)$. Let p be an arbitrary element of A. If $p \in \mathcal{T}_\Sigma$, then $p \in A \cap \mathcal{T}_\Sigma = L(p_1) \cup \ldots \cup L(p_m)$, so there exists a number j such that $t_j = p$. Thus, p is enumerated by step 8 of **Procedure 4**. If not, then there exists a set F that satisfies the condition of step 11 by Lemma 5.8(1). Let n_0 be the least n satisfying $\{t_1, \ldots, t_n\} \supset F$. It is clear that p is enumerated at Step 11 when $n = n_0$.

We end this section by giving an example.

Example 5.1. Suppose $\Sigma = \{a, b, f, g\}, rank(a) = rank(b) = 0, rank(f) = 2, rank(g) = 1$, and $x, y \in V$. Let us consider learning $L(f(a, x)) \cup L(f(x, b)) \in \cup^{\leq 2}\mathcal{TPL}$. Let a positive presentation σ be as follows:

$$t_1 = f(a, a),\ t_2 = f(a, f(a, b)),\ t_3 = f(b, b),$$
$$t_4 = f(a, g(a)),\ t_5 = f(a, b),\ t_6 = f(g(a), b), \ldots$$

This time **Procedure 4** learns $L(f(a, x)) \cup L(f(x, b))$ as follows:

•$n = 1$: $\mathcal{GPD}$ outputs t_1 and **Procedure 4** outputs (t_1).
•$n = 2$: $\mathcal{GPD}$ outputs t_2 and **Procedure 4** outputs (t_1, t_2).

•$n = 3$: $lca(t_1, t_2) = f(a, x)$, $lca(t_1, t_3) = f(x, x)$, $lca(t_2, t_3) = f(x, y)$. Hence $\mathcal{GPD}$ outputs only t_3. **Procedure 4** chooses two larger elements from $\{t_1, t_2, t_3\}$ and output them.
•$n = 4$: Since $lca(t_1, t_2) = lca(t_1, t_4) = lca(t_2, t_4) = f(a, x)$, $\mathcal{GPD}$ outputs t_4 and $f(a, x)$. **Procedure 4** outputs two maximal elements of $\{t_1, \ldots, t_4, f(a, x)\}$, that is, $f(a, x)$ and t_3.
•$n = 5$: Since $lca(t_1, t_5) = lca(t_2, t_5) = lca(t_4, t_5) = f(a, x)$, $\mathcal{GPD}$ outputs t_5 and $f(a, x)$. **Procedure 4** outputs $f(a, x)$ and the larger element of $\{t_3, t_5\}$.
•$n = 6$: Since $lca(t_3, t_5) = lca(t_3, t_6) = lca(t_5, t_6) = f(x, b)$, $\mathcal{GPD}$ outputs t_6 and $f(x, b)$. **Procedure 4** outputs $(f(a, x), f(x, b))$.

6 Conclusions

We have seen that the notion of characteristic set and its computability play important role to give a learning procedure of bounded unions of Noetherian closed set systems. The existence of characteristic set has not been used to give a concrete learning procedure. This is probably because the existence of a characteristic set for each language is weaker condition than finite thickness or finite elasticity. Also both finite thickness and finite elasticity are properties concerning family of language, while the existence of a characteristic set just depends on each language. The point of our paper is to put emphasis on a characteristic set and to show that it is useful for certain classes of languages.

Acknowledgment

This research is partially supported by Grant-in-Aid 19300046 from JSPS. The second author is partially supported by K18-XI-234 from Kayamori Foundation of Informational Science Advancement.

References

1. Angluin, D.: Inductive Inference of Formal Languages from Positive Data. Information and Control 45, 117–135 (1980)
2. Arimura, H., Shinohara, T., Otsuki, S.: Polynomial Time Inference of Unions of Tree Pattern, Languages. In: Proc. the 2nd Workshop on Algorithmic Learning Theory (ALT 1991), pp. 105–114 (1991)
3. Arimura, H., Shinohara, T., Otsuki, S.: A Polynomial Time Algorithm for Finding Finite Unions of Tree Patterns. In: Brewka, G., Jantke, K.P., Schmitt, P.H. (eds.) NIL 1991. LNCS, vol. 659, pp. 118–131. Springer, Heidelberg (1993)
4. de Brecht, M., Kobayashi, M., Tokunaga, H., Yamamoto, A.: Inferability of Closed Set Systems From Positive Data. In: Washio, T., Satoh, K., Takeda, H., Inokuchi, A. (eds.) JSAI 2006. LNCS (LNAI), vol. 4384, pp. 265–275. Springer, Heidelberg (2007)
5. Hayashi, S.: Mathematics Based on Learning. In: Cesa-Bianchi, N., Numao, M., Reischuk, R. (eds.) ALT 2002. LNCS (LNAI), vol. 2533, pp. 7–21. Springer, Heidelberg (2002)

6. Kapur, S., Bilardi, G.: On uniform learnability of language families. Information Processing Letters 44, 35–38 (1992)
7. Kobayashi, S.: Approximate Identification, Finite Elasticity and Lattice Structure of Hypothesis Space, Technical Report, CSIM 1996-2004, Dept. of Compt. Sci. and Inform. Math., Univ. of Electro-Communications (1996)
8. Lassez, J.L., Maher, M.J., Marriott, K.: Unification Revisited. In: Minker, J. (ed.) Foundations of Deductive Databases and Logic Programming, pp. 587–626. Morgan-Kaufman, San Francisco (1988)
9. Stephan, F., Ventsov, Y.: Learning Algebraic Structures from Text. Theoretical Computer Science 268, 221–273 (2001)
10. Takamatsu, I., Kobayashi, M., Tokunaga, H., Yamamoto, A.: Computing Characteristic Sets of Bounded Unions of Polynomial Ideals. In: Satoh, K., Inokuchi, A., Nagao, K., Kawamura, T. (eds.) JSAI 2007. LNCS (LNAI), vol. 4914, pp. 318–329. Springer, Heidelberg (2008)
11. Tokunaga, H., Yamamoto, A.: Inductive Inference of Bounded Unions of Languages with Complete Characteristic Sets (in Japanese), SIG-FPAI-A703-01, 01-08 (2008)
12. Wright, K.: Identification of Unions of Languages Drawn from an Identifiable Class. In: Proc. of COLT 1989, pp. 328–388. Morgan-Kaufman, San Francisco (1989)