A Sufficient Condition for Learning Unbounded Unions of Languages with Refinement Operators

Tomohiko Okayama, Ryo Yoshinaka, Keisuke Otaki and **Akihiro Yamamoto**
Graduate School of Informatics, Kyoto University
Yoshida Honmachi, Sakyo-ku, Kyoto, Japan 606-8501

Abstract

This paper presents a natural sufficient condition on a class of languages under which all the unions of any number of languages from the class are learnable from positive examples (data) in the Gold-style. Learning unions of languages models information extraction from mixed data from different sources. The Gold-style learning has provided many fruitful results on learning unions of bounded number of languages, while few positive results on learning unions of unbounded number of languages has been known. In this research, we focus on a condition of the class of languages on which refinement operators are defined. Refinement operators are fundamental tools to transform a hypotheses, which represents a language, into a set of hypotheses, which represent a subsets of the language.

1 Introduction

In this research, we present a sufficient condition on a class of languages under which all the unions of any number of languages from the class are learnable from positive data. Our central research subject is learnability of unions of languages in the Gold-style learning.

The Gold-style learning from positive data, called *identification in the limit from positive data* is popular in the field of computational learning theory (Angluin, 1980; Gold, 1967). Based on the rough observation of learning natural languages, Gold proposed a learning model called *identification in the limit from positive data* :

- A learner receives a datum belonging to a target language step by step, infinitely many times.
- A learner outputs a hypothesis each time receives a datum.

In this paper, we adopt the Gold-style learning for learning a class of unions of languages. Let X be an enumerable set of objects. A language is any subset of X. A language mapping L maps a *hypothesis* h to a language, denoted by $L(h)$. The set of hypotheses $\mathcal{H}$ is called *hypothesis space*. The class of languages defined by $\mathcal{H}$ is $\mathcal{C} = \{ L(h) \subseteq X \mid h \in \mathcal{H} \}$.

In the process of learning unions of languages in the Gold-Style, a learner receives mixed data from different languages and constructs a set of hypotheses representing a union of languages. In the learning, the number of languages consisting a union is often bounded. If the upper bound is k, the class of up to k unions of languages $\mathcal{C}^k = \{L(h_1) \cup \cdots \cup L(h_n) | \exists n \in \mathbb{N}^+, n \leq k\}$ is called (k-)*bounded unions of languages*. The class of every finite union of languages $\mathcal{C}^* = \{L(h_1) \cup \cdots \cup L(h_n) | n \in \mathbb{N}^+\}$, which is called *unbounded unions of languages*.

Many classes of bounded unions of languages are shown to be learnable from positive data (Jain, Ng, and Tay, 2001). Some of these results is obtained thanks to Wright (1989). He proved that a class of languages with a property, called *finite elasticity* (Wright, 1989; Motoki, Shinohara, and Wright, 1991), is learnable from positive data. He also showed that if a class of languages has finite elasticity, the class of k-bounded unions of these languages is learnable from positive data. The following motivating example shows that learnability of any bounded unions of languages does not imply the learnability of unbounded unions of languages.

Example 1. For any set A, let the number of elements of A be $|A|$. Let the hypothesis space be $\mathcal{H} = \{\langle 0 \rangle, \langle 1 \rangle, ...\} \cup \{\langle * \rangle\}$, and a language mapping be

$$L(\langle n \rangle) = \{n\} \text{ for } \langle n \rangle \in \{\langle 0 \rangle, \langle 1 \rangle, ...\}$$
$$L(\langle * \rangle) = \mathbb{N}.$$

The class of languages is $\mathcal{C} = \{L(\langle n \rangle) \subseteq \mathbb{N} \mid \langle n \rangle \in \mathcal{H}\} = \{\{0\}, \{1\}, ...\} \cup \{\mathbb{N}\}$. Then, for $k \in \mathbb{N}^+$, the class of k-bounded unions of languages is $\mathcal{C}^k = \{ S \mid S \subseteq \mathbb{N}, |S| \leq k \} \cup \{ \mathbb{N} \}$ and the class of unbounded unions of languages is $\mathcal{C}^* = \{ S \mid S \subseteq \mathbb{N}, |S| < \infty \} \cup \{ \mathbb{N} \}$. Now we check the learnabilty of $\mathcal{C}^k$. Let I be any set of integers in a target union of languages in $\mathcal{C}^k$. If $|I|$ is not more than k for any I, the set of hypotheses is $\{\langle i \rangle \in \mathcal{H} \mid i \in I\}$, which represents a target union of languages. Otherwise the set of hypotheses is $\{\langle * \rangle\}$. It is clear that $\mathcal{C}^k$ is learnable from positive data. However, $\mathcal{C}^*$ is not learnable from positive data because of a learnability result provided by Gold. Gold showed that any class of formal languages that contains every finite language and at least one infinite language, called *super finite* is not identifiable from positive data. $\mathcal{C}^*$ consists of all finite subsets of $\mathbb{N}$ and one $\mathbb{N}$. $\mathbb{N}$ is an infinite languages. Therefore $\mathcal{C}^*$ is super finite.

Additionally, there are few clues to find learnable classes of unbounded number of languages (Jain, Ng, and Tay, 2001).

Our main result is to prove that a class of unbounded unions of languages is learnable if a refinement operator on the class of these languages satisfies Condition 3 (Section 2) and the class of these languages satisfies Condition 12 (Section 3), not the class of unbounded unions of these languages. A refinement operator transforms a hypothesis h into other hypotheses $g_1, ..., g_n$ such that $L(g_i) \subseteq L(h)$ $(1 \leq i \leq n)$. In our proof, we construct a refinement operator $\tilde{\rho}$ on the class of unbounded unions of languages based on a refinement operator on the class of these languages, and show that $\tilde{\rho}$ also satisfies Condition 3 to prove the learnability of the class of unbounded unions of these languages.

The rest of this paper is organized as follows. In Section 2, we introduce the Gold-style learning and learning with refinement operators by Ouchi and Yamamoto. In Section 3, we show details of unbounded unions of languages, and show proofs for learning unbounded unions with a refinement operator. In Section 4, we conclude this paper.

2 Preliminaries

Let $\mathbb{N}$, $\mathbb{N}^+$ and $\mathbb{Q}$ be the sets of non-negative integers, positive integers and retional numbers, respectively.

Gold-style Learning from Positive Data

Let X be a recursively enumerable set of *objects*. A $language$ is a subset of X. Let $\mathcal{H}$ be a recursively enumerable set of *hypothesis*, called a *hypothesis space*. Let $L(\cdot)$, called *language mapping*, be a mapping from $\mathcal{H}$ to 2^X, where there is a recursive function $f : X \times \mathcal{H} \rightarrow \{0, 1\}$ such that $f(w, h) = 1$ if and only if $w \in L(h)$. The class of languages is defined as $\mathcal{C} = \{L(h) \subseteq X \mid h \in \mathcal{H}\}$. A triplet $(\mathcal{C}, \mathcal{H}, L(\cdot))$ is called *a concept class*. The Gold-style learning (1967), *identification in the limit from positive data* is a learning model in which a learner, often called an *inductive inference machine*, *IIM* receives examples from a learning target language $C \in \mathcal{C}$ one by one. Each time *IIM* receives a new example, it outputs a hypothesis $h \in \mathcal{H}$ as a conjecture that may represent the target language. We would like the conjecture to converge to a hypothesis h that indeed represents $C = L(h)$. More formally, A *positive presentation* of a language $C \in \mathcal{C}$ is an infinite sequence $\sigma = \langle \sigma[1], \sigma[2], ... \rangle$ of objects in which all and only elements of C occur. We denote by σ_n, the prefix of σ of length n, i.e., $\sigma_n = \langle \sigma[1], ..., \sigma[n] \rangle$. Each object in σ is called a (*positive*) *example*. *IIM*(σ_n) denotes n-th outout of the *IIM* which receives σ. If there is $N \in \mathbb{N}^+$ such that $\forall i \geq N, [$*IIM*$(\sigma_i) =$ *IIM*$(\sigma_N)]$, we say the *IIM converges to IIM*(σ_N) on σ. We say that the *IIM identifies a language C in the limit from positive data* if *IIM* converges to h such that $L(h) = C$ on any positive presentation σ of C. For a concept class $(\mathcal{C}, \mathcal{H}, L(\cdot))$, if an *IIM* identifies C in the limit from positive data for all $C \in \mathcal{C}$, we say that an *IIM identifies $\mathcal{C}$ in the limit from positive data*. A concept class $(\mathcal{C}, \mathcal{H}, L(\cdot))$ is *identifiable from positive data* if and only if there exists an *IIM* that identifies the class $\mathcal{C}$ in the limit from positive data.

Procedure Learn-with-Refinement-and-MINL($\mathcal{H}, \rho$)
Input: A positive presentation $\sigma = \langle e_1, e_2, ... \rangle$ of C, a mapping L from $\mathcal{H}$ to $\mathcal{C}$.
Output: An enumeration of $\mathcal{H}$

```
S := ∅, L(h_0) := ∅
for i = 1 to ∞ do
    S := S ∪ {e_i}
    if e_i ∈ L(h_i) then
        h_i := h_{i−1}
    else if MINL(T, S, i) = "no hypothesis" then
        h_i := h_{i−1}
    else
        h_i := MINL(T, S, i)
    end if
    output h_i as a guess
end for
```

Algorithm MINL(T, S, n)
Input: A finite set T in Theorem 4, any finite subset S of any language C, and a natural number n.
Output: A hypothesis in $\mathcal{H}$ or "no hypothesis"

```
H_0 := T
for j = 0 to n do
    if ∃h ∈ H_j.[S ⊆ L(h) ∧ ∀g ∈ ρ(h).S ⊈ L(g)] then
        return the hypothesis h which satisfies the condition above and is firstly enumerated
    end if
    H_{j+1} := {g ∈ H | g ∈ ρ(H_j) ∧ S ⊆ L(g)}
end for
return "no hypothesis"
```

Figure 1: Learning with the MINL strategy and a refinement operator (Ouchi and Yamamoto, 2010).

Learning from Positive Data with Refinement Operators

In this section, we review the work of Ouchi and Yamamoto (2010) concerning learning from positive data with refinement operators. Refinement operators were firstly introduced by Shapiro for learning logic programs (1981) to search for hypotheses efficiently. Laird (1988) adapted refinement operators for learning a class of languages from positive and negative examples. Ouchi and Yamamoto modified the definition of refinement operators and provided learning procedure Learn-with-Refinement-and-MINL($\mathcal{H}$,ρ) and an algorithm MINL(T, S, n) in Figure 1. Then, they proved that classes of languages are learnable from positive data if a refinement operator satisfies Condition 3. The definition of their refinement operators are formally defined as follows.

Definition 2 (Ouchi and Yamamoto (2010))**.** Let $(\mathcal{C}, \mathcal{H}, L(\cdot))$ be a concept class. A mapping $\rho : \mathcal{H} \rightarrow 2^{\mathcal{H}}$ is called a *refinement operator* on the class if it satisfies the following three condition:

[R-1] For every $h \in \mathcal{H}$, $\rho(h)$ is recursively enumerable.

[R-2] For every $h \in \mathcal{H}$, $g \in \rho(h) \Rightarrow L(g) \subseteq L(h)$.

[R-3] There exists no sequence $h_1, h_2, ..., h_n$ of hypotheses such that $h_1 = h_n$ and $h_{i+1} \in \rho(h_i)$ $(1 \le i \le n-1)$.

For any $H \subseteq \mathcal{H}$, we define $\rho(H) = \{h' \in \mathcal{H} \mid \exists h \in H, [h' \in \rho(h)]\}$. $\rho^k(h)$ is defined for each $k \in \mathbb{N}$ as $\rho^0(h) = \{h\}$, and $\rho^{k+1}(h) = \rho(\rho^k(h))$. We also define $\rho^*(h) = \bigcup_{k \in \mathbb{N}} \rho^k(h)$, and $\rho^+(h) = \bigcup_{k \in \mathbb{N}^+} \rho^k(h)$, respectively.

Ouchi and Yamamoto proved that every concept class is identifiable in the limit from positive data if it admits a refinement operator satisfying the next conditions.

Condition 3. Let ρ be a refinement operator on $(\mathcal{C}, \mathcal{H}, L(\cdot))$.

[A-1] ρ is *locally finite*, which means ρ is a computable function.

[A-2] ρ is *semantically complete*, which means that, for every $h \in \mathcal{H}$ and every $C \in \mathcal{C}$ such that $C \subsetneq L(h)$, there exists a hypothesis h' such that $L(h') = C$ and $h' \in \rho^+(h)$, in other words, there exists a finite sequence $h_1, h_2, ..., h_n \in \mathcal{H}$ such that $h_1 = h$, $L(h_n) = C$ and $h_{i+1} \in \rho(h_i)$ $(1 \le i \le n-1)$.

[A-3] There is a finite set $T \subseteq \mathcal{H}$ such that $\mathcal{C} = \{\, L(h) \mid h \in \bigcup_{t \in T} \rho^*(t) \,\}$. We call T an *initial hypothesis set*.

[A-4] There is no infinite sequence $h_1, h_2, ... \in \mathcal{H}$ such that $h_{i+1} \in \rho(h_i)$ and $L(h_i) = L(h_{i+1})$ for all $i \ge 1$.

Note that [A-3] is slightly weaker than the original.

Theorem 4 (Ouchi and Yamamoto 2010)**.** If a refinement operator on a concept class satisfies Condition 3, the concept class is identifiable in the limit from positive data.

In the proof of Theorem 4, Ouchi and Yamamoto showed that the learning procedure Learn-with-Refinement-and-MINL($\mathcal{H}, \rho$) illustrated in Figure 1 works for correct learning.

Example 5. Let

$$\mathcal{H}_0 = \{\langle m, n\rangle \mid m, n \in \mathbb{N}^+\},$$
$$L(\langle m, n\rangle) = \{(x, y) \in \mathbb{Q} \times \mathbb{Q} \mid m \le x, n \le y\}, \text{ and}$$
$$\mathcal{C}_0 = \{L(\langle m, n\rangle) \mid m, n \in \mathbb{N}^+\}.$$

Clearly $(\mathcal{C}_0, \mathcal{H}_0, L(\cdot))$ is a concept class. Let $\rho_0(\langle m, n\rangle) = \{\langle m+1, n\rangle, \langle m, n+1\rangle\}$ be a mapping : $\mathcal{H} \to 2^{\mathcal{H}}$ and $T = \{\langle 1, 1\rangle\}$ as the initial hypothesis set. Then, ρ_0 is a refinement operator and satisfies [A-1] to [A-4]. Thus $(\mathcal{C}_0, \mathcal{H}_0, L(\cdot))$ is identifiable from positive data.

3 Learning Unbounded Unions of Languages with a Refinement Operator

In this section, we prove that the class of unbounded unions of languages is learnable from positive data if $(\mathcal{C}, \mathcal{H}, L(\cdots))$ satisfies Condition 12 and admits a refinement operator satisfying Condition 3.

Bounded Unions of Languages

Definition 6 (Bounded union)**.** For a positive k and a concept class $(\mathcal{C}, \mathcal{H}, L(\cdot))$, the concept class of k-bounded unions of languages $(\mathcal{C}^k, \mathcal{H}^k, L(\cdot))$ is defined as

$$\begin{aligned} \mathcal{H}^k &= \{\{h_1, ..., h_i\} \mid 1 \le i \le k\}, \\ L(\{h_1, ..., h_n\}) &= L(h_1) \cup \cdots \cup L(h_n), \\ \mathcal{C}^k &= \{L(H) \mid H \in \mathcal{H}^k\}. \end{aligned}$$

for some positive integer k.

Motoki, Shinohara, and Wright (1989; 1991) presented a property of the class of languages, called *finite elasticity*. Wright (1989) proved that if a class of languages has finite elasticity, it is learnable from positive data. He also proved the following proposition:

Proposition 7 (Wright 1989)**.** If a class of languages $\mathcal{C}$ has finite elasticity, then the class of up to k unions of languages $\mathcal{C}^k$, called bounded unions of languages, also has finite elasticity for some positive integer k.

That means once a language class $\mathcal{C}$ is shown to have finite elasticity, the class of bounded unions of languages has also finite elasticity and thus is learnable.

Proposition 7 gives many rich identifiable classes of languages. However, the class of unbounded unions of languages is beyond the scope of this proposition (See the Example 1).

Unbounded Unions of Languages

Some studies tackled the learning of unbounded unions of languages. Shinohara and Arimura (2000) does the first important work for unbounded unions of languages. They showed that if a class of languages satisfies a certain strong condition in addition to finite elasticity, the class of unbounded unions of languages will have finite elasticity and thus be the learnable from positive data.

However, there are few classes of unbounded unions of languages which are proved to be identifiable from positive data using their results. In this section, we provide another approach to show learnability of unbounded unions of languages.

Definition 8 (Unbounded union)**.** For a concept class $(\mathcal{C}, \mathcal{H}, L(\cdot))$, the concept class of unbounded unions of languages $(\mathcal{C}^*, \mathcal{H}^*, L(\cdot))$ is defined as

$$\begin{aligned} \mathcal{H}^* &= \{\{h_1, ..., h_i\} \mid i \in \mathbb{N}^+\}, \\ L(\{h_1, ..., h_n\}) &= L(h_1) \cup \cdots \cup L(h_n), \\ \mathcal{C}^* &= \{L(H) \mid H \in \mathcal{H}^*\}. \end{aligned}$$

Note that any set of hypotheses in $\mathcal{H}^*$ has finite number of elements.

Example 9 (Example 5 continued)**.** For $H = \{\langle 1, 3\rangle, \langle 2, 2\rangle, \langle 4, 1\rangle\} \in \mathcal{H}^*$, the language $L(H) = L(\langle 1, 3\rangle) \cup L(\langle 2, 2\rangle) \cup L(\langle 4, 1\rangle)$ is illustrated in Figure 2.

Lemma 10 (Kanazawa 1998)**.** Let $C_1, C_2, ...$ be an infinite sequence of languages in $\mathcal{C}$. If it satisfies $C_1 \subsetneq C_2 \subsetneq \cdots$ and $\bigcup_{i \in \mathbb{N}^+} C_i \in \mathcal{C}$, then the concept class is not identifiable from positive data.

The following motivating example shows that the existence of a refinement operatpr ρ on a concept class $(\mathcal{C}, \mathcal{H}, L(\cdot))$ satisfying Condition 3 does not imply the learnability of the concept class $(\mathcal{C}^*, \mathcal{H}^*, L(\cdot))$.

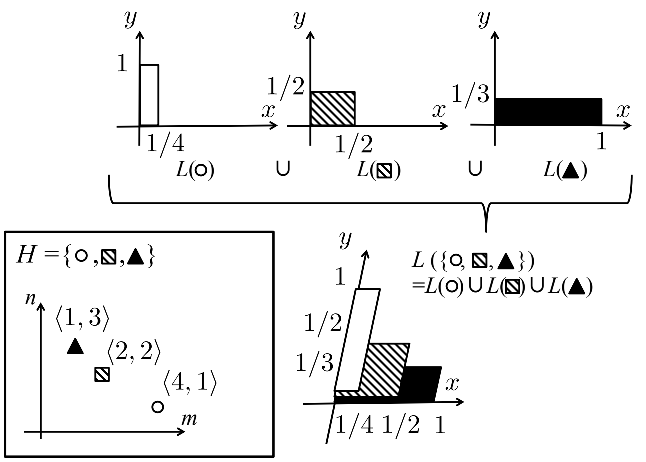

Figure 2: Unions of languages in Example 9.

Example 11 (Ouchi and Yamamoto 2010)**.** Let a hypothesis space be $\mathbb{N}^+$, and a language mapping be

$$L(n) = \begin{cases} \{x \in \mathbb{Q} \mid 0 \leq x \leq 1\} & \text{if } n = 0, \\ \left\{x \in \mathbb{Q} \mid 0 \leq x \leq 1 + \frac{1}{m}\right\} & \text{if } n = 2m-1, \text{and} \\ \left\{x \in \mathbb{Q} \mid 0 \leq x \leq 1 + \frac{1}{m}\right\} & \\ \setminus \left\{\frac{1}{m+k} \in \mathbb{Q} \mid k \in \mathbb{N}^+\right\} & \text{if } n = 2m. \end{cases}$$

for some $m \in \mathbb{N}$, for all $n \in \mathbb{N}^+$. A refinement operator on the class is defined as: $\forall n \in \mathbb{N}, \exists m \in \mathbb{N}^+$,

$$\rho(n) = \begin{cases} \emptyset & \text{if } n = 0 \\ \{2m, 2m+1, 0\} & \text{if } n = 2m-1 \\ \emptyset & \text{if } n = 2m \end{cases}$$

for all $n \in \mathbb{N}^+$. Let an initial hypothesis set be $T = \{1\}$. Then this refinement operator satisfies [A-1] to [A-4]. The concept class $(\mathcal{C}, \mathcal{H}, L)$ is identifiable from positive data by Theorem 4.

Consider the concept class $(\mathcal{C}^*, \mathcal{H}^*, L(\cdot))$. Let $D_n = \bigcup_{k=1}^{n} C_{2k} \in \mathcal{C}$. Then $D_n \subsetneq D_{n+1}$ for all $n \in \mathbb{N}^+$. Moreover, $\bigcup_{n \in \mathbb{N}^+} D_n = C_1$. From Lemma 10, the concept class $(\mathcal{C}^*, \mathcal{H}^*, L)$ is not identifiable from positive data.

Condition 12. Let $(\mathcal{C}, \mathcal{H}, L(\cdot))$ be a concept class.

[C-1] $L(\cdot)$ is bijective.

[C-2] For any $p, q_1, q_2, ..., q_n \in \mathcal{H}$,

$$L(p) \subseteq L(q_1) \cup L(q_2) \cup \cdots \cup L(q_n) \Leftrightarrow \exists i (1 \leq i \leq n), [L(p) \subseteq L(q_i)].$$

Note that [C-2] is called *compactness with respect to containment* of $\mathcal{C}$ (Arimura, Shinohara, and Otsuki, 1994).

Lemma 13. Let $(\mathcal{C}, \mathcal{H}, L(\cdot))$ be a concept class satisfying [C-1] and it admit a refinement operator ρ satisfying [A-2].

$$\forall h \in \mathcal{H}, [\forall g \in \mathcal{H}, [L(g) \subseteq L(h) \Leftrightarrow g \in \rho^*(h)]]$$

Proof. From [C-1], for any p and q in $\mathcal{H}$,

$$L(g) = L(h) \Leftrightarrow g = h \Leftrightarrow g \in \rho^0(h)$$

holds. From [A-2] and [C-1], for any p and q in $\mathcal{H}$,

$$L(g) \subsetneq L(h) \Leftrightarrow g \in \rho^+(h)$$

holds. Therefore,

$$L(g) \subseteq L(h) \Leftrightarrow g \in \rho^*(h)$$

□

Lemma 14. Let $(\mathcal{C}, \mathcal{H}, L(\cdot))$ be a concept class satisfying [C-1] and ρ be a refinement operator on $(\mathcal{C}, \mathcal{H}, L(\cdot))$ satisfying [A-1] to [A-2]. For, $g, h \in \mathcal{H}$, whether g belongs to $\rho^*(h)$ is decidable.

Proof. We present an algorithm which decides whether $g \in \rho^*(h)$. It repeats the following procedure for $k = 1, 2, \ldots$ until it finds a proof for or a counterexample to $g \in \rho^*(h)$. Let $e_1, e_2, \ldots$ be an enumeration of the objects of X. In the kth repetition, the algorithm checks whether $g \in \rho^k(h)$, which is computable by [A-1]. If so, it outputs "true" and halts. Otherwise, if $e_k \in L(g) \setminus L(h)$, it outputs "false" and halts.

If $g \in \rho^*(h)$, which means $L(g) \subseteq L(h)$, there is no $e \in L(g) \setminus L(h)$, thus it never outputs "false". Since there is k such that $g \in \rho^k(h)$ from [A-2], at the kth repetition, it outputs "true". If $g \notin \rho^*(h)$, which means $L(g) \not\subseteq L(h)$, there is $e_k \in X$ such that $e_k \in L(g) \setminus L(h)$. □

Example 15 (Example 11 continued)**.** $(\mathcal{C}^*, \mathcal{H}^*, L)$ does not have [C-2], because $C_6 \subseteq C_4 \cup C_8$ does not mean $C_6 \subseteq C_4$ or $C_6 \subseteq C_8$.

Example 16 (Example 5 continued)**.** ρ_0 on $\mathcal{H}_0$ satisfies

$$\langle m, n\rangle \in \rho_0^*(\langle m', n'\rangle) \text{ if and only if } m' \leq m \wedge n' \leq n.$$

Then

$$\begin{aligned} L(\langle m, n\rangle) \subseteq L(\langle m', n'\rangle) &\Leftrightarrow m' \leq m \wedge n' \leq n \\ &\Leftrightarrow \langle m, n\rangle \in \rho_0^*(\langle m', n'\rangle). \end{aligned}$$

That means [C-1] holds. Let $L(\langle m, n\rangle) \subseteq L(\langle m_1, n_1\rangle) \cup \cdots \cup L(\langle m_k, n_k\rangle)$ for some $k \in \mathbb{N}^+$. From $\langle m, n\rangle \in L(\langle m, n\rangle)$, there exists i $(1 \leq i \leq k)$ such that $\langle m, n\rangle \in L(\langle m_i, n_i\rangle)$.

$$\begin{aligned} \langle m, n\rangle \in L(\langle m_i, n_i\rangle) &\Leftrightarrow m_i \leq m \wedge n_i \leq n \\ &\Leftrightarrow \langle m, n\rangle \in \rho_0^*(\langle m_i, n_i\rangle). \end{aligned}$$

This means that [C-2] holds.

Proving Learnability of Unbounded Unions of Languages with a Refinement Operator on the Class

We prove that a concept class $(\mathcal{C}^*, \mathcal{H}^*, L(\cdot))$ is identifiable from positive data when $(\mathcal{C}, \mathcal{H}, L(\cdot))$ satisfies condition [A-1] to [A-4] and [C-1] to [C-2]. In the following, we deal with concept classes satisfying conditions [A-1] to [A-4] and [C-1]-[C-2].

Lemma 17.

$$\forall P, Q \in \mathcal{H}^*.[L(P) \subseteq L(Q) \Leftrightarrow \forall p \in P.[\exists q \in Q.[p \in \rho^*(q)]].$$

Proof. "If" part clearly holds. From [C-2], there is $q \in Q$ such that $L(p) \subseteq L(q)$ for any $p \in P$. From [C-1] and Lemma 13, "only if" part holds. □

Definition 18 (reduce operator)**.** A mapping $reduce : \mathcal{H}^* \to \mathcal{H}^*$ called the *reduce operator* is defined by

$$reduce(P) = \{p \in P \mid \forall q \in P \setminus \{p\}, [p \notin \rho^+(q)]\}.$$

If $P = reduce(P)$, P is called *reduced*. Note that $reduce(\cdot)$ is computable because of Lemma 14.

Lemma 19. Let P be a finite subset of $\mathcal{H}$. The following always holds.

$$L(P) = L(reduce(P)).$$

Proof. Let $Q = P \setminus reduce(P)$. Because of the definition of the reduce operator, $\forall q \in Q, [\exists p \in reduce(P), [q \in \rho^+(p)]]$. From Lemma 17, $L(Q) \subseteq L(reduce(P))$ holds. Therefore, $L(P) = L(reduce(P)) \cup L(Q) = L(reduce(P))$. □

Definition 20. Let the concept class of languages be $(\mathcal{C}^*, \mathcal{H}^*, L(\cdot))$. We define

$$\mathcal{H}^*_{red} = \{H \in \mathcal{H}^* \mid H = reduce(H)\}.$$

We define a map $\tilde{\rho}$ based on ρ for the unbounded unions of languages. It will be shown that $\tilde{\rho}$ is indeed a refinement operator on $(\mathcal{C}^*, \mathcal{H}^*, L(\cdot))$ and moreover satisfies the conditions [A-1] to [A-4].

Definition 21. We define a new mapping $\tilde{\rho} : \mathcal{H}^* \to 2^{\mathcal{H}^*}$ as follows:

$$\tilde{\rho}(P) = \bigcup_{p \in reduce(P)} \Big(\{P \cup \rho(p) \setminus \{p\}\}\Big) \cup \bigcup_{p \in P} \Big(\{P \setminus \{p\}\}\Big).$$

Lemma 22. Let $P \in \mathcal{H}^*$ and $Q = P \cup \rho(p) \setminus \{p\}$ for some $p \in reduce(P)$. Then,

$$L(Q) \subsetneq L(P)\,.$$

Proof. We have $L(\rho(p)) \subseteq L(p)$ by [R-2]. Therefore

$$L(Q) \subseteq L(P) \cup L(\rho(p)) = L(P)\,.$$

To derive a contradiction, suppose that $L(Q) = L(P)$. By [C-2], there is $q \in Q$ such that $L(p) \subseteq L(q)$, which implies $p \in \rho^*(q)$ by [C-1] and Lemma 13. Since $p \notin Q$, we have $p \neq q$, i.e., $p \in \rho(q)$. The fact $p \in reduce(P)$ implies $q \notin P$. Then q must be from $\rho(p)$, which means $L(q) \subseteq L(p)$ by [R-2]. This contradicts $p \in \rho(q)$. □

Lemma 23. The following two holds.

- the mapping $\tilde{\rho}$ of Definition 21 satisfies the conditions [R-1]-[R-3] of refinement operators, and
- [A-1] and [A-4] on $(\mathcal{C}^*, \mathcal{H}^*, L(\cdot))$ holds.

Proof. **[A-1]** Let $P = \{p_1, p_2, ..., p_n\} \in \mathcal{H}^*$ and $P' = reduce(P) = \{p'_1, ..., p'_m\}$. Let $Q_i = P \setminus \{p_i\}$ $(1 \le i \le n)$ and $Q_{i+n} = P \cup \rho(p'_i) \setminus \{p'_i\}$ $(1 \le i \le m)$, respectively. Then $\tilde{\rho}(P) = \{Q_1, ..., Q_n, Q_{n+1}, ..., Q_{m+n}\}$, that is, $\tilde{\rho}(P)$ is finite and enumerable, and $\tilde{\rho}$ is computable because $reduce(\cdot)$ is computable.

[R-1] It immediately follows from [A-1].

[R-2] If $Q \in \tilde{\rho}(P)$, either
(1) $Q = P \setminus \{p\}$ for some $p \in P$, or
(2) $Q = (P \setminus \{p\}) \cup \rho(p)$ for some $p \in reduce(P)$.
In the case (1), $L(Q) = L(P \setminus \{p\}) \subseteq L(P)$. In the case (2), $L(Q) \subsetneq L(P)$ from Lemma 22.

[A-4] To derive a contradiction, suppose that there was an infinite sequence of hypotheses $P_1, P_2, ... \in \mathcal{H}^*$ such that $P_{i+1} \in \tilde{\rho}(P_i)$ and $L(P_i) = L(P_{i+1})$ for all $i \ge 1$. Lemma 22 implies $P_{i+1} = P_i \setminus \{p_i\}$ for some $p_i \in P_i$. That is, $P_1 \supsetneq P_2 \supsetneq P_3 \supsetneq \dots$, which is impossible by the finiteness of P_1.

[R-3] Suppose [R-3] does not hold. There exists a finite sequence $P_1, ..., P_n \in \mathcal{H}^*$ such that $P_{i+1} \in \tilde{\rho}(P_i)$ for all i $(1 \le i \le n-1)$ and $P_1 = P_n$. From [R-2], we have $L(P_1) \supseteq \cdots \supseteq L(P_n) \supseteq L(P_1)$. That is, $L(P_1) = \cdots = L(P_n) = L(P_1)$. This is impossible by [A-4]. □

Lemma 24 (Semantical completeness of $\tilde{\rho}$)**.**

$$\forall P \in \mathcal{H}^*, [\forall Q \in \mathcal{H}^*_{red}, [L(Q) \subseteq L(P) \Rightarrow Q \in \tilde{\rho}^*(P)]]$$

holds. Therefore, [A-2] on $(\mathcal{C}^*, \mathcal{H}^*, L(\cdot))$ also holds by Lemma 19.

Proof. Let $P \in \mathcal{H}^*$ and $Q \in \mathcal{H}^*_{red}$ be such that $L(Q) \subseteq L(P)$. Lemma 17 implies $\forall q \in Q, \exists p \in P, [q \in \rho^*(p)]$, that is, $Q \subseteq \rho^*(P)$.

For $H \in \mathcal{H}^*$ and $h \in \rho^*(H)$, we define

$$\delta(h, H) = \min\{\, k \in \mathbb{N} \mid \exists h' \in H, h \in \rho^k(h')\,\}$$

and for $H' \subseteq \rho^*(H)$,

$$\delta(H', H) = \sum_{h \in H'} \delta(h, H)\,.$$

We show by induction on $\delta(Q, P)$ that $Q \subseteq \rho^*(P)$ implies $Q \in \tilde{\rho}^*(P)$.

Basis: If $\delta(Q, P) = 0$, we have $Q \subseteq P$. Clearly $Q \in \tilde{\rho}^*(P)$.

Induction Step: Suppose $\delta(Q, P) > 0$. Let f be a function from Q to P such that $\delta(q, f(q)) = \delta(q, P)$ for all $q \in Q$. Let

$$P_f = \{\, f(q) \in P \mid q \in Q\,\} \setminus Q\,,$$
$$P' = (P \cap Q) \cup P_f\,.$$

Since P' is a subset of P,

$$P' \in \tilde{\rho}^*(P). \quad (1)$$

Note that P_f is not empty because $\delta(Q, P) \neq 0$. Let p_{max} be an element of $reduce(P_f)$. Because Q is reduced, p_{max} is also in $reduce(P')$. We let

$$P'' = (P' \cup \rho(p_{max})) \setminus \{p_{max}\} \in \tilde{\rho}^*(P')\,. \quad (2)$$

For each $q \in Q$ we have

$$\begin{cases} \delta(q, P'') = \delta(q, P) - 1 & \text{if } f(q) = p_{max}, \\ \delta(q, P'') \le \delta(q, P) & \text{otherwise}\,. \end{cases}$$

Indeed there is $q \in Q$ such that $f(q) = p_{max}$. Therefore,

$$\delta(Q, P'') < \delta(Q, P) .$$

By the induction hypothesis,

$$Q \in \tilde{\rho}^*(P'') \tag{3}$$

holds. By (1), (2) and (3) we finally obtain

$$Q \in \tilde{\rho}^*(P) . \qquad \square$$

Lemma 25. [A-3] on $(\mathcal{C}^*, \mathcal{H}^*, L(\cdot))$ holds.

Proof. For every $C \in \mathcal{C}^*$, there is $Q \in \mathcal{H}^*_{red}$ such that $L(Q) = C$ by Lemma 19. By [A-3] and [C-1] on ρ, we have $Q \subseteq \rho^*(T)$ for an initial hypothesis set T for $(\mathcal{C}, \mathcal{H}, L(\cdot))$. Lemma 24 implies $Q \in \tilde{\rho}^*(T)$, that is, $\{T\}$ is an initial hypothesis set for $(\mathcal{C}^*, \mathcal{H}^*, L(\cdot))$. $\square$

Finally, we achieve the following theorem, that is, a new positive result of learnability of the unbounded unions of languages.

Theorem 26 (Main Result)**.** Let $(\mathcal{C}, \mathcal{H}, L(\cdot))$ be a concept class which admits a refinement operator ρ satisfying Condition 3 and satisfies Condition 12 The concept class $(\mathcal{C}^*, \mathcal{H}^*, L(\cdot))$ is identifiable from positive data.

Proof. From Lemmas 23, 24, and 25, the refinement operator $\tilde{\rho}$ satisfies [A-1]-[A-4] on $(\mathcal{C}^*, \mathcal{H}^*, L(\cdot))$. From Theorem 4, $(\mathcal{C}^*, \mathcal{H}^*, L(\cdot))$ is identifiable from positive data. $\square$

Comparing Our Result with Other Learnability Result

There are some sufficient conditions for learnability of classes of languages from positive data. We introduce one of them, Condition 27, and compare it with our result.

Condition 27. Let $(\mathcal{C}, \mathcal{H}, L(\cdot))$ be a concept class such that $L(\cdot)$ satisfies [C-1]. The concept class $(\mathcal{C}, \mathcal{H}, L(\cdot))$ satisfies Condition 27 if for any $h \in \mathcal{H}$, there exists a finite set $S_h \subseteq L(h)$ such that

- S_h is finite,
- $S_h \subseteq L(h)$, and
- $\forall h' \in \mathcal{H}, [S_h \subseteq L(h') \Rightarrow L(h) \subseteq L(h')]$.

We say that a concept class has *characteristic sets* if it satisfies Condition 27.

Theorem 28. Let $(\mathcal{C}, \mathcal{H}, L(\cdot))$ be a concept class. The concept class $(\mathcal{C}, \mathcal{H}, L(\cdot))$ is identifiable from positive data if $(\mathcal{C}, \mathcal{H}, L(\cdot))$ has characteristic sets.

It is known that if a concept class has finite elasticity as we mentioned in Section 3, the concept class has characteristic sets. We show that our result does not imply Condition 27 by presenting a non-trivial learnable class of unbounded unions of languages. This example is inspired by Ouchi and Yamamoto (2010).

Example 29. Let a hypothesis space be $\mathbb{N}$ and

$$L(n) = \begin{cases} \{\, x \in \mathbb{Q} \mid 0 \le x \le 1 \,\} & \text{if } n = 0 \\ \left\{\, x \in \mathbb{Q} \;\middle|\; 0 \le x \le 1 + \frac{1}{m} \,\right\} & \text{if } n = 2m-1 \\ \left\{\, x \in \mathbb{Q} \;\middle|\; 0 \le x \le 1 + \frac{1}{m} \,\right\} & \\ \setminus \left\{\, \frac{1}{m+k} \in \mathbb{Q} \;\middle|\; k \in \mathbb{N}^+ \,\right\} & \\ \setminus \left\{\, 1 + \frac{1}{m+\frac{k}{2}} \in \mathbb{Q} \;\middle|\; k \in \mathbb{N}^+ \,\right\} & \text{if } n = 2m \end{cases}$$

for some $m \in \mathbb{N}^+$ for all $n \in \mathbb{N}$.

A refinement operator ρ on the class is defined as : $\forall n \in \mathbb{N}, \exists m \in \mathbb{N}^+$,

$$\rho(n) = \begin{cases} \emptyset & \text{if } n = 0 \\ \{2m, 2m+1, 0\} & \text{if } n = 2m-1 \\ \emptyset & \text{if } n = 2m \end{cases}$$

Let $T = \{1\}$. Then ρ clearly satisfies [A-1] to [A-4]. $L(\cdot)$ on the concept class clearly satisfies [C-1]. Now we show the concept class satisfies [C-2].

Suppose that

$$L(2m) \subseteq \bigcup_{k \in I} L(k)$$

for some $m \in \mathbb{N}^+$ and finite set $I \subseteq \mathbb{N}$. The fact $1 + 1/m \in L(2m)$ implies that there is $k \in I$ such that $1 + 1/m \in L(k)$, which means

$$k \in \{1, 3, \ldots, 2m-1\} \cup \{2m\} .$$

For any $k \in \{1, 3, \ldots, 2m-1\}$ clearly $L(2m) \subseteq L(k)$.

Suppose that

$$L(2m-1) \subseteq \bigcup_{k \in I} L(k)$$

for some $m \in \mathbb{N}^+$ and finite set $I \subseteq \mathbb{N}$. The fact $1 + \frac{1}{m+1/2} \in L(2m)$ implies that there is $k \in I$ such that $1 + \frac{1}{m+1/2} \in L(k)$. This means that

$$k \in \{1, 3, \ldots, 2m-1\} ,$$

for which we have $L(2m-1) \subseteq L(k)$.

Suppose that

$$L(0) \subseteq \bigcup_{k \in I} L(k)$$

for some finite set $I \subseteq \mathbb{N}$. Let $m = \max I$. The fact $\frac{1}{m+1} \in L(0)$ implies that there is an odd number k in I, for which we have $L(0) \subseteq L(k)$.

Hence, the concept class satisfies [C-2].

Here we show that no finite set can be a characteristic set of $L(0)$. For a finite set $S \subseteq L(0)$, let

$$m = \left\lceil \frac{1}{\min S} \right\rceil .$$

Then we have $S \subseteq L(2m)$ and $L(0) \not\subseteq L(2m)$.

4 Concluding Remarks

We have proved that the concept class $(\mathcal{C}^*, \mathcal{H}^*, L(\cdot))$ is identifiable from positive data with the refinement operator $\tilde{\rho}$ if a refinement operator ρ on $(\mathcal{C}, \mathcal{H}, L(\cdot))$ satisfying [A-1] to [A-4] and $(\mathcal{C}, \mathcal{H}, L(\cdot))$ satisfies Condition 12.

The Theorem 26 in this paper is a generalization of Theorem 4 in (Ouchi and Yamamoto, 2010): they showed a learnable class, called a class of unbounded unions of *constant-free regular tree pattern languages*. Indeed their target class satisfies the conditions [C-1] and [C-2], although their refinement operator for the class of unions of these langauges is different from ours. A *tree* is a term or an atom in formal logic, and a *tree languages* is the set of trees which are the ground instances of a tree pattern.

Arimura et al. (1994) presented an efficient way, called *minimal multiple generalization*, to learn heads of definite clauses of a logic program from positive data. To obtain several definite clauses of the program, it is important to divide a set of positive data into subsets and construct *least common generalizations* from these subsets. Arimura, Shinohara, and Otsuki (1991) gives a way to divide positive data into bounded number of divisions. Ishizaka, Arimura, and Takeshi (1993) pointed out that learning tree pattern languages is applicable to know heads of definite clauses, and Arimura et al. (1994) shows that sets of bounded number of definite clauses without bodies, called *unit clauses*, are learnable from positive data. Learning unbounded unions of constant-free regular tree pattern languages (Ouchi and Yamamoto, 2010) means that a class of finite sets of logic programs that consists of unit clauses are learnable from positive data.

In our future work, we will give new learnable classes of unbounded unions of languages, not only formal languages but also logic programs using our result.

References

Angluin, D. 1980. Finding patterns common to a set of strings. *Journal of Computer and System Sciences* 21(1):46–62.

Arimura, H.; Shinohara, T.; Otsuki, S.; and Ishizaka, H. 1994. A generalization of the least general generalization. In *Machine Intelligence 13*, 59–85. Oxford University Press.

Arimura, H.; Shinohara, T.; and Otsuki, S. 1991. A polynomial time algorithm for finding finite unions of tree pattern languages. In *Proceedings of the Second International Workshop on Nonmonotonic and Inductive Logic*, 118–131. Springer-Verlag.

Arimura, H.; Shinohara, T.; and Otsuki, S. 1994. Finding minimal generalizations for unions of pattern languages and its application to inductive inference from positive data. In *Proceedings of the 11th Annual Symposium on Theoretical Aspects of Computer Science*, 649–660. Springer-Verlag.

de la Higuera, C. 2010. *Grammatical inference: learning automata and grammars*. Cambridge University Press.

Gold, E. M. 1967. Language identification in the limit. *Information and control* 10(5):447–474.

Ishizaka, H.; Arimura, H.; and Takeshi, S. 1993. Inductive learning of logic programs from positive facts using minimal multiple generalization (in japanese). *Journal of the Japanese Society for Artificial Intelligence* 8(4):419–426.

Jain, S.; Ng, Y.; and Tay, T. 2001. Learning languages in a union. In Abe, N.; Khardon, R.; and Zeugmann, T., eds., *Algorithmic Learning Theory*, volume 2225 of *Lecture Notes in Computer Science*. Springer Berlin Heidelberg. 235–250.

Kanazawa, M. 1998. *Learnable Classes of Categorial Grammars*. Studies in Logic, Language and Information. The Center for the Study of Language and Information Publications.

Laird, P. D. 1988. *Learning from good and bad data*. Norwell, MA, USA: Kluwer Academic Publishers.

Motoki, T.; Shinohara, T.; and Wright, K. 1991. The correct definition of finite elasticity: corrigendum to identification of unions. In *Proceedings of the fourth annual workshop on Computational learning theory*, COLT '91, 375–. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc.

Ouchi, S., and Yamamoto, A. 2010. Learning from positive data based on the minl strategy with refinement operators. In Nakakoji, K.; Murakami, Y.; and McCready, E., eds., *New Frontiers in Artificial Intelligence*, volume 6284 of *Lecture Notes in Computer Science*. Springer Berlin Heidelberg. 345–357.

Shapiro, E. Y. 1981. Inductive inference of theories from facts. *Research Report YALEU/DCS/RR-192, Yale University*.

Shinohara, T., and Arimura, H. 2000. Inductive inference of unbounded unions of pattern languages from positive data. *Theoretical Computer Science* 241(1):191–209.

Wright, K. 1989. Identification of unions of languages drawn from an identifiable class. In *Proceedings of the second annual workshop on Computational learning theory*, COLT '89, 328–333. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc.