JMLR: Workshop and Conference Proceedings 217:8–22, 2023 ICGI regular paper

Extending Distributional Learning from Positive Data and Membership Queries

Makoto Kanazawa KANAZAWA@HOSEI.AC.JP
Hosei University

Ryo Yoshinaka RYOSHINAKA@TOHOKU.AC.JP
Tohoku University

Editors: François Coste, Faissal Ouardi and Guillaume Rabusseau

Abstract

We consider an extension of distributional learning of context-free languages (from positive data and membership queries), where nonterminals are represented by extended regular expressions (allowing all Boolean operations) augmented by atoms corresponding to membership queries. These nonterminals classify a string based not just on its distribution, but also on the distributions of its substrings. The learning algorithm for this extension works in essentially the same way as in previous works on distributional learning, while targeting a significantly larger class of context-free languages.

Keywords: distributional learning; membership queries; context-free grammars; extended regular closure; star-free closure

1. Introduction

Distributional learning of context-free languages from positive data and membership queries (Clark and Yoshinaka, 2016; Kanazawa and Yoshinaka, 2017, 2021) roughly works as follows.[1] When forming a new hypothesis from a set T of positive examples, the learner first constructs a set of nonterminals using fragments of strings in T as building blocks. The total number of copies of these building blocks that may be used in constructing a nonterminal is bounded by a global parameter, but the size of each building block is constrained only by the available positive data and is potentially unbounded. Each nonterminal B constructed by the learner is supposed to "denote" a set $[\![B]\!]^{L_*}$ of strings, relative to the target language L_*. The denotation of a nonterminal is defined in such a way that the learner can determine whether a given string x belongs to $[\![B]\!]^{L_*}$ in polynomial time (in the length of x and the description size of B) by making queries to the membership oracle for L_*.

With the set of nonterminals at hand, the learner then considers candidate productions involving these nonterminals, each of the form

$$B_0 \to w_0\, B_1\, w_1\, \ldots\, B_n\, w_n\,, \tag{1}$$

where terminal strings $w_0, w_1, \ldots, w_n$ are again fragments of strings in the positive data—to be precise, they are nonoverlapping substrings occurring in that order in some string in T. The number n of right-hand side nonterminals is bounded by a global parameter, so that given positive data T, the set of candidate productions is finite (and its description size is

1. Of the two approaches to distributional learning, we only consider the *dual* version in this paper.

polynomial in the total lengths of the strings in T). The production (1) is *valid* (relative to L_*) if and only if

$$[\![B_0]\!]^{L_*} \supseteq w_0\ [\![B_1]\!]^{L_*}\ w_1\ \ldots\ [\![B_n]\!]^{L_*}\ w_n\,. \tag{2}$$

The learner's goal is to retain only those productions that are valid, but since the denotation of a nonterminal is in general infinite, the learner performs an approximate test of validity:

$$[\![B_0]\!]^{L_*} \supseteq w_0\,(\mathrm{Sub}(T) \cap [\![B_1]\!]^{L_*})\,w_1\ \ldots\ (\mathrm{Sub}(T) \cap [\![B_n]\!]^{L_*})\,w_n\,. \tag{3}$$

Here, $\mathrm{Sub}(T)$ is the set of all strings that occur as substrings in some element of T. The context-free grammar G hypothesized by the learner consists of all productions that pass the test (3).

When a new positive example t arrives, the learner first sees whether $T \cup \{t\}$ is included in the language of the current hypothesis G. When this is so, the nonterminals of G are kept intact, and the approximate test of validity is performed with the now-enlarged set $\mathrm{Sub}(T \cup \{t\})$ in place of $\mathrm{Sub}(T)$. The effect is that some productions previously ruled in may now be ruled out, resulting in a new grammar G' with a smaller set of productions. When $T \cup \{t\}$ is not included in the language of G, then a set of nonterminals is built afresh and the above procedure is repeated (with $T \cup \{t\}$ in place of T).

On top of this rough outline, previous distributional learning algorithms made certain specific decisions about the building blocks out of which nonterminals are built and how they may be combined, together with how the constructed nonterminals are interpreted. In all previous works (Clark and Yoshinaka, 2016; Kanazawa and Yoshinaka, 2017, 2021), building blocks are nonoverlapping prefix-suffix pairs (u, v) of some string in the positive data. These pairs of strings each denote a *quotient* of L_*:[2]

$$u\backslash L_*/v = \{\, x \mid uxv \in L_* \,\}\,.$$

A nonterminal consists of a collection of such string pairs together with a certain Boolean operation on sets; its denotation is the corresponding Boolean combination of the quotients. The Boolean operations were restricted to finite intersections in earlier papers on distributional learning (Clark and Yoshinaka, 2016; Kanazawa and Yoshinaka, 2017), but the restriction was lifted by Kanazawa and Yoshinaka (2021)—in other words, the denotation (relative to L_*) of a nonterminal constructed by the learner may be any set in the Boolean closure of the quotients of L_*.[3]

In this paper, we consider an extension of distributional learning which still follows the rough outline we just gave above but which no longer deserves to be called purely "distributional". In this new approach, a nonterminal is a *relativized extended regular expression*[4] built from $\varnothing$, ε, terminal symbols, and *query atoms* $(u, v)^{\triangleleft}$ (denoting quotients $u\backslash L_*/v$). Extended regular expressions allow all Boolean operations in addition to concatenation and Kleene star. The possible denotations of nonterminals correspond to the *extended regular*

2. In the literature, the quotient $u\backslash L_*/v$ is often written $u^{-1}L_*v^{-1}$.
3. There is in fact a caveat; see Section 3.2 below.
4. The use of the term *extended regular expression* to refer to regular expressions augmented with the symbol for complement (and intersection) seems common, but note that the term is sometimes used to refer to regular expressions extended in other ways, for instance with the device for back reference.

closure of the quotients of L_*, the smallest set containing the basic regular sets $\varnothing$, $\{\varepsilon\}$, $\{c\}$ ($c \in \Sigma$) and the quotients of L_* that is closed under the regular and Boolean operations.

If a set K is in the Boolean closure of the quotients of L_*, whether a string x belongs to K only depends on the syntactic congruence class of x (its "distribution"). Thus, when $\{\,(u,v) \mid uxv \in L_*\,\} = \{\,(u,v) \mid uyv \in L_*\,\}$, we have $x \in K$ if and only if $y \in K$. This is no longer so if concatenation or Kleene star is used in forming K from quotients of L_*. Membership of x in K will then depend not only on the distribution of x but also on the distributions of its substrings. Not only that, if K is in the extended regular closure of the quotients of L_*, membership of x in K may even depend on the exact identity of x, since the extended regular closure of the quotients of L_* contains all regular sets, including singletons.

Thus, in the extended setting of the present paper, whether a string belongs to the denotation of a nonterminal is not completely determined by its distribution—the string's internal makeup matters. Despite this rather drastic departure from previous approaches, the learning algorithm and the proof of its correctness remain essentially unchanged. Unsurprisingly, the class of context-free languages that can be targeted by the new algorithm (with global parameters set to some values) contain languages that were previously out of reach of distributional learning. We also give an example of a rather simple context-free language that cannot be targeted even by our new algorithm (with any choice of global parameters).

The present paper is our first look at this new class of context-free languages. We currently know very little about its boundary, as well as the "landscape" inside it. We mention one outstanding question that suggests itself toward the end of the paper (Section 5), which concerns the indispensability of the Kleene star operation in the representation of nonterminals.

2. Basic Definitions and Properties

2.1. Extended Regular Closure

Let $\mathcal{L} \subseteq \mathscr{P}(\Sigma^*)$. We say that $\mathcal{L}$ is closed under an operation $g\colon (\mathscr{P}(\Sigma^*))^k \to \mathscr{P}(\Sigma^*)$ if $g(L_1, \ldots, L_k) \in \mathcal{L}$ holds for every $L_1, \ldots, L_k \in \mathcal{L}$. If Γ is a set of operations on $\mathscr{P}(\Sigma^*)$ (of any arity), then the Γ-*closure* of $\mathcal{L}$ is the smallest superclass of $\mathcal{L}$ that is closed under each operation in Γ. If Γ is the set consisting of the operations of union, intersection, and complement (relative to Σ^*), Γ-closure is called *Boolean closure*, and if Γ in addition contains the operations of concatenation and Kleene star, as well as the "nullary" operations $\varnothing$, $\{\varepsilon\}$, $\{c\}$ ($c \in \Sigma$), then Γ-closure is called *extended regular closure*.

We say that $L \subseteq \Sigma^*$ is *closed under substring* if $x \in L$ implies that all substrings of x are in L. The following is an important property of extended regular closure.

Lemma 1 *Suppose that R is a regular set that is closed under substring. If K is in the extended regular closure of $\mathcal{L}$, then $K \cap R$ is in the extended regular closure of $\{\,L \cap R \mid L \in \mathcal{L}\,\}$.*

Proof The proof is by induction on K (or more precisely, on the evidence that K is in the extended regular closure of $\mathcal{L}$).

- $K \in \mathcal{L}$. Then $K \cap R \in \{ L \cap R \mid L \in \mathcal{L} \}$.
- $K \in \{\varnothing, \{\varepsilon\}\} \cup \{ \{c\} \mid c \in \Sigma \}$. Then $K \cap R$ is either $\varnothing$ or K itself.
- $K = K_1 \cup K_2$. Then $K \cap R = (K_1 \cup K_2) \cap R = (K_1 \cap R) \cup (K_2 \cap R)$. By the induction hypothesis applied to K_1 and K_2, both $K_1 \cap R$ and $K_2 \cap R$ are in the extended regular closure of $\{ L \cap R \mid L \in \mathcal{L} \}$, which implies that $K \cap R$ is as well.
- $K = K_1 \cap K_2$. Similar to the case of union.
- $K = \overline{K_1}$. Then $K \cap R = \overline{K_1} \cap R = \overline{K_1 \cap R} \cap R$. By the induction hypothesis, $K_1 \cap R$ is in the extended regular closure of $\{ L \cap R \mid L \in \mathcal{L} \}$. Since R is regular, $K \cap R$ is in the extended regular closure of $\{ L \cap R \mid L \in \mathcal{L} \}$.
- $K = K_1 K_2$. Since R is closed under substring, $K \cap R = K_1 K_2 \cap R = (K_1 \cap R)(K_2 \cap R) \cap R$. By the induction hypothesis applied to K_1 and K_2, both $K_1 \cap R$ and $K_2 \cap R$ are in the extended regular closure of $\{ L \cap R \mid L \in \mathcal{L} \}$. Since R is regular, it follows that $K \cap R$ is in the extended regular closure of $\{ L \cap R \mid L \in \mathcal{L} \}$.
- $K = K_1^*$. Since R is closed under substring, $K \cap R = K_1^* \cap R = (K_1 \cap R)^* \cap R$. By the induction hypothesis, $K_1 \cap R$ is in the extended regular closure of $\{ L \cap R \mid L \in \mathcal{L} \}$. Since R is regular, it follows that $K \cap R$ is in the extended regular closure of $\{ L \cap R \mid L \in \mathcal{L} \}$. ■

2.2. Sound Pre-fixed Points of Context-Free Grammars

We adopt the standard definition of a *context-free grammar* (CFG):[5] A CFG is a 4-tuple $G = (N, \Sigma, P, S)$, where N is a finite set of *nonterminals*, Σ is a finite set of terminals, S is a member of N called the *start symbol*, and P is a finite subset of $N \times (N \cup \Sigma)^*$. Members of P are called *productions* and are written in the form $B_0 \to w_0\, B_1\, w_1 \ldots B_n\, w_n$, where the B_i are nonterminals and the w_i are strings of terminals. We take for granted the derivation relation $\Rightarrow_G^*$ on $(N \cup \Sigma)^*$. The *language* of a nonterminal B is $L_G(B) = \{ x \in \Sigma^* \mid B \Rightarrow_G^* x \}$, but we often write B in place of $L_G(B)$ for brevity. The *language* of G is $L(G) = L_G(S)$. A nonterminal B is *reachable* if $S \Rightarrow_G^* uBv$ for some $(u, v) \in \Sigma^* \times \Sigma^*$.

A tuple $(X_B)_{B \in N}$ of subsets of Σ^* is a *pre-fixed point* of G if for each production $B_0 \to w_0\, B_1\, w_1 \ldots B_n\, w_n$ of G, it holds that

$$X_{B_0} \supseteq w_0\, X_{B_1}\, w_1 \ldots X_{B_n}\, w_n\,.$$

It is well known that the tuple $(L_G(B))_{B \in N}$ consisting of the set of strings derived from each nonterminal is the least pre-fixed point of G under the partial order of componentwise inclusion. Needless to say, a CFG in general has many pre-fixed points; the tuple $(X_B)_{B \in N}$ with $X_B = \Sigma^*$ for all B is always a pre-fixed point. We say that $(X_B)_{B \in N}$ is a *sound pre-fixed point* (SPP) of G if it is a pre-fixed point and $X_S \subseteq L(G)$ (or equivalently, $X_S = L(G)$). The least pre-fixed point is of course always an SPP, but a CFG in general can have many SPPs (Kanazawa and Yoshinaka, 2017).

5. An alternative is to allow a set of *initial nonterminals*, instead of a single start symbol, a convention which is useful for the *primal* version of distributional learning algorithms. The difference does not matter for the purpose of this paper.

If $(X_B)_{B\in N}$ is a pre-fixed point of G, then we can prove by induction that $A \Rightarrow_G^* w_0\, B_1\, w_1 \ldots B_n\, w_n$ implies $X_A \supseteq w_0\, X_{B_1}\, w_1 \ldots X_{B_n}\, w_n$. So we have

Lemma 2 *Let $G = (N, \Sigma, P, S)$ be a CFG and let $L = L(G)$. If $(X_B)_{B\in N}$ is an SPP of G and $S \Rightarrow_G^* uBv$, then $X_B \subseteq u\backslash L/v$.*

We say that a context-free grammar for a language L has the Γ-*closure property* if it has an SPP consisting of sets in the Γ-closure of $\{\, u\backslash L/v \mid (u, v) \in \Sigma^* \times \Sigma^* \,\}$. The *very weak finite context property* of Kanazawa and Yoshinaka (2017) is the intersection closure property (the Γ-closure property for the case $\Gamma = \{\cap\}$). The class of CFGs with the Boolean closure property corresponds to the class $\bigcup_{k,l,m} \mathbf{FCP}(k, l, m)$ of CFLs discussed by Kanazawa and Yoshinaka (2021).

Example 1 Let $L = \{\, a^m b^n \mid 1 \leq m \wedge (m = n \vee 2m \leq n) \,\}$. The following context-free grammar G generates L:

$$S \to S_1 \mid S_2, \quad S_1 \to ab \mid aS_1b, \quad S_2 \to abb \mid aS_2bb \mid S_2b.$$

Writing B for $L_G(B)$ for brevity, we have

$$\begin{aligned} S &= L = \varepsilon\backslash L/\varepsilon, \\ S_1 &= \{\, a^m b^n \mid 1 \leq m = n \,\} \\ &= L \cap \overline{a\backslash L/bb}, \\ S_2 &= \{\, a^m b^n \mid 1 \leq m \wedge 2m \leq n \,\} \\ &= L \cap a\backslash L/bb, \end{aligned}$$

which shows that G has the Boolean closure property.

We can show that G does not have the intersection closure property. Suppose that X_{S_1} is the S_1-component of an SPP of G. We have $S \Rightarrow_G^* a^l S_1 b^l$ for all $l \geq 0$, so by Lemma 2, we must have $X_{S_1} \subseteq a^l\backslash L/b^l$ for all $l \geq 0$. If $a^m b^n \in X_{S_1}$, then $a^{m+2n} b^{3n} \in L$, which implies $m = n$. This shows that $X_{S_1} = S_1$. It is easy to see that if $S_1 \subseteq u\backslash L/v$, then we must have $u = a^l$, $v = b^l$ for some l. But for every $l \in \mathbb{N}$, $b^n \in a^l\backslash L/b^l$ whenever $l \leq n$. This shows that S_1 cannot be a finite intersection of sets of the form $a^l\backslash L/b^l$.

Applying the pumping lemma to a sufficiently long string of the form $a^n b^n$, we can show that any grammar for L must have a nonterminal like S_1. So L does not have a CFG with the intersection closure property.

3. Learning

3.1. Relativized Extended Regular Expressions

Evidence that a set K is in the Γ-closure of the quotients of a language L can be expressed by a certain expression formed with symbols standing for the operations in Γ. In order for the kind of learning algorithm we are interested in to target context-free grammars with the Γ-closure property, such an expression must translate into a polynomial-time reduction (by which K reduces to L). As in previous works in distributional learning, the reduction in question for the case of the extended regular closure property is a special kind of *polynomial-time truth-table reduction* (Ladner et al., 1974).

The set $\mathcal{R}$ of *relativized extended regular expressions* is defined inductively as follows:

- $\varnothing \in \mathcal{R}$.
- $\varepsilon \in \mathcal{R}$.
- $c \in \mathcal{R}$ for each $c \in \Sigma$.
- $(u,v)^\triangleleft \in \mathcal{R}$ for each $(u,v) \in \Sigma^* \times \Sigma^*$.
- If $e_1 \in \mathcal{R}$ and $e_2 \in R$, then the following are all in $\mathcal{R}$:

$$e_1 \cup e_2\,, \quad e_1 \cap e_2\,, \quad \overline{e_1}\,, \quad e_1 e_2\,, \quad e_1^*\,.$$

We refer to an expression of the form $(u,v)^\triangleleft$ as a *query atom*. Let the *size* of $e \in \mathcal{R}$ be the number of (occurrences of) subexpressions of e (which is equal to the number of occurrences of atomic expressions and Boolean and regular operations in e).

If $e \in \mathcal{R}$, then its *denotation* $[\![e]\!]^L$ relative to a language $L \subseteq \Sigma^*$ is defined as follows:

$$\begin{aligned}
[\![\varnothing]\!]^L &= \varnothing, & [\![e_1 \cup e_2]\!]^L &= [\![e_1]\!]^L \cup [\![e_2]\!]^L\,, & [\![e_1 e_2]\!]^L &= [\![e_1]\!]^L\,[\![e_2]\!]^L\,,\\
[\![\varepsilon]\!]^L &= \{\varepsilon\}, & [\![e_1 \cap e_2]\!]^L &= [\![e_1]\!]^L \cap [\![e_2]\!]^L\,, & [\![e_1^*]\!]^L &= ([\![e_1]\!]^L)^*.\\
[\![c]\!]^L &= \{c\}, & [\![\overline{e_1}]\!]^L &= \Sigma^* - [\![e_1]\!]^L\,, & &\\
[\![(u,v)^\triangleleft]\!]^L &= u\backslash L/v, & & & &
\end{aligned}$$

Let $e \in \mathcal{R}$, and let $C_e = \{\,(u,v) \mid (u,v)^\triangleleft \text{ occurs in } e\,\}$. For $x \in \Sigma^*$, let $\mathrm{Sub}(x)$ denote the finite set of all substrings of x. Then $x \in [\![e]\!]^L$ iff $x \in [\![e]\!]^L \cap \mathrm{Sub}(x)$, and the proof of Lemma 1 should make it fairly clear that the truth value of "$x \in [\![e]\!]^L$" is some truth function of the truth values of "$uyv \in L$" for all combinations of $y \in \mathrm{Sub}(x)$ and $(u,v) \in C_e$.

A Boolean circuit for this truth function can be constructed from e and x in polynomial time. The circuit has input nodes for some of the pairs $((u,v),y)$, where $(u,v) \in C_e$ and $y \in \mathrm{Sub}(x)$, representing the truth value of "$uyv \in L$". It has gates for some of the pairs (e',y) consisting of a subexpression e' of e that is not a query atom and a string $y \in \mathrm{Sub}(x)$, representing the truth value of "$y \in [\![e']\!]^L$". If e' is one of the atomic regular expressions $\varnothing$, ε, or c ($c \in \Sigma$), the gate for (e',y) has fan-in zero, representing the constant truth value $\top$ or $\bot$, depending on whether y is in the set denoted by e'. If $e' = e_1 e_2$ and $|y| = n$, then $n+1$ intermediate AND gates of fan-in 2 are created, corresponding to the $n+1$ different ways of splitting y into strings y_1, y_2 such that $y = y_1 y_2$. These intermediate gates each represent the truth value of $y_1 \in [\![e_1]\!]^L \wedge y_2 \in [\![e_2]\!]^L$. The gate for $(e_1 e_2, y)$ is then an OR gate of fan-in $n+1$ representing their disjunction. For (e_1^*, y), if $y = \varepsilon$, then the gate for it is the constant $\top$. Otherwise, it is an OR of n AND gates, corresponding to $n = |y|$ ways of splitting y into y_1 and y_2 such that $y_1 \neq \varepsilon$ and $y = y_1 y_2$; each of these n gates represents the truth value of $y_1 \in [\![e_1]\!]^L \wedge y_2 \in [\![e_1^*]\!]^L$. The gates for other types of subexpression, $e_1 \cup e_2, e_1 \cap e_2, \overline{e_1}$, are simple OR, AND, and NOT gates of fan-in 2, 2, and 1, respectively. The total number of gates and inputs of the circuit is linear in the size of e and (at most) cubic in $|x|$, and its fan-in is bounded by $\max\{|x|+1, 2\}$, so it can be evaluated in time that is linear in the size of e and polynomial in $|x|$.

Example 2 Let $e = (aa, bb)^\triangleleft \cap \overline{((a,b)^\triangleleft\, \overline{\varnothing})(a \mid b)}$. The Boolean circuit for $ab \in [\![e]\!]^L$ is shown in Figure 1. Input nodes are represented by rectangles. (This circuit is equivalent to the formula $aa(ab)bb \in L \wedge (a\varepsilon b \notin L \wedge aab \notin L)$.)

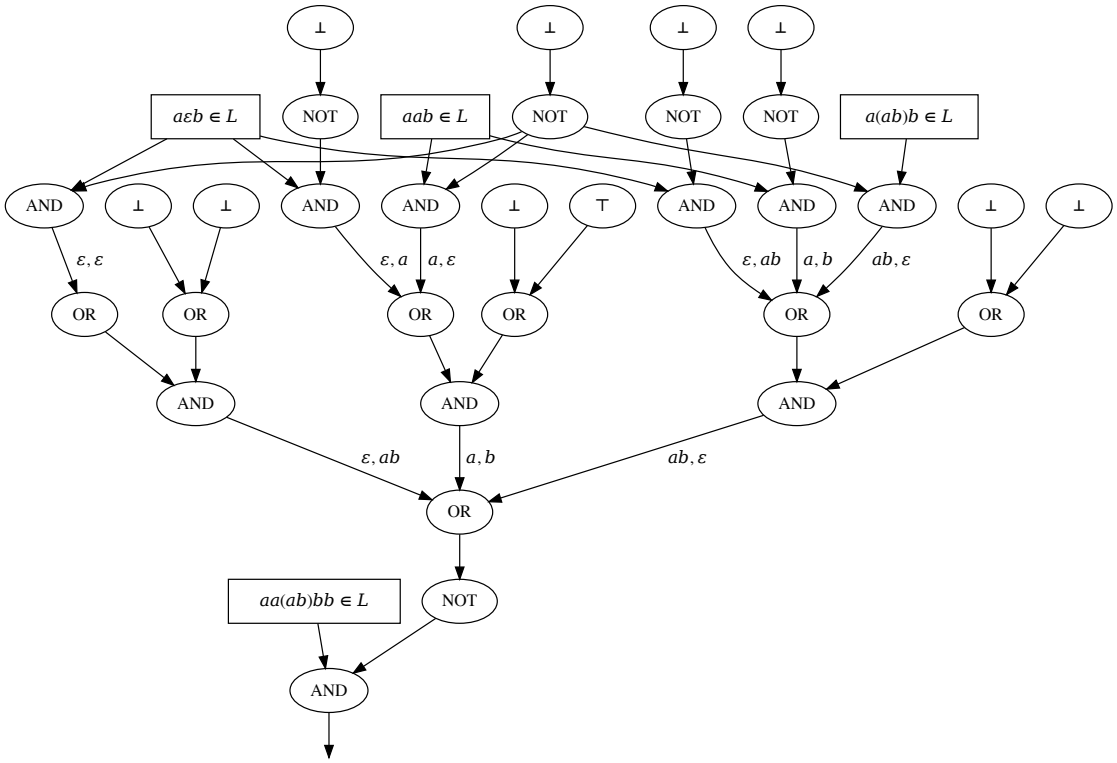

Figure 1: The Boolean circuit for $ab \in \left[\!\left[(aa, bb)^{\triangleleft} \cap \overline{((a, b)^{\triangleleft}\, \overline{\varnothing})(a \mid b)}\right]\!\right]^{L}$.

3.2. Algorithm

Let **EXREG** be the class of context-free languages that have a grammar with the extended regular closure property. Just like in previous works (Kanazawa and Yoshinaka, 2017, 2021), we need to restrict **EXREG** by a couple of parameters to obtain a class that can be learned by an algorithm similar to the previous (dual) learners.

We write $\mathbf{EXREG}_r(k)$ for the class of context-free languages L that have a grammar $G = (N, \Sigma, P, S)$ satisfying the following properties:

- the right-hand side of each production in P has at most r nonterminals, and
- there is a *guarded* expression $e_B \in \mathcal{R}$ with $\mathrm{size}(e_B) \leq k$ for each $B \in N$ such that $(\llbracket e_B \rrbracket^L)_{B \in N}$ is an SPP of G.

The meaning of "guarded" is explained below.

At each stage, the learner for $\mathbf{EXREG}_r(k)$ uses as nonterminals relativized extended regular expressions of size at most k involving query atoms $(u, v)^{\triangleleft}$ such that (u, v) is a nonoverlapping prefix-suffix pair found in the positive data. An important necessary restriction that was not adequately emphasized by Kanazawa and Yoshinaka (2021) is that every nonterminal B hypothesized by the learner must satisfy the property $\llbracket B \rrbracket^{L_*} \subseteq \mathrm{Sub}(L_*)$, where L_* is the target language and $\mathrm{Sub}(K) = \bigcup\{\, \mathrm{Sub}(x) \mid x \in K \,\}$. This is necessary for

the approximate test of validity (3) to guarantee validity (2) in the limit. The following inductive definition of a *guarded* relativized extended regular expression is a simple way of ensuring this property:

- $(u, v)^\triangleleft$ is guarded.
- If e_1 is guarded, so is $e_1 \cap e_2$. (The expression e_2 need not be guarded.)
- If e_1 and e_2 are guarded, so is $e_1 \cup e_2$.

When e is guarded, its *guards* are the query atoms in e that witness the fact that e is guarded. In the following lemma, C is the set of all pairs (u, v) such that $(u, v)^\triangleleft$ occurs as a guard in e.

Lemma 3 *If e is guarded, then there is a finite set $C \subseteq \Sigma^* \times \Sigma^*$ such that for every $L \subseteq \Sigma^*$,*

$$[\![e]\!]^L \subseteq \bigcup_{(u,v)\in C} u\backslash L/v\,.$$

In particular, $[\![e]\!]^L \subseteq \mathrm{Sub}(L)$.

Using only guarded expressions as nonterminals is not restrictive in the following sense:

Lemma 4 *Let $G = (N, \Sigma, P, S)$ be a CFG with the extended regular closure property such that every nonterminal of G is reachable. Then there is a guarded relativized extended regular expression e_B for each $B \in N$ such that $([\![e_B]\!]^L)_{B\in N}$ is an SPP of G.*

Proof If $([\![e_B]\!]^L)_{B\in N}$ is an SPP of G and $S \Rightarrow_G^* u_B B v_B$ for each $B \in N$, then Lemma 2 implies $[\![e_B]\!]^L = [\![(u_B, v_B)^\triangleleft \cap e_B]\!]^L$. ■

We use the following notation:

$$\begin{aligned}
\mathrm{Sub}(T) &= \{\, x \in \Sigma^* \mid uxv \in T \text{ for some } u, v \,\},\\
\mathrm{Sub}^n(T) &= \{\, (w_1, \ldots, w_n) \in (\Sigma^*)^n \mid u_0 w_1 u_1 \ldots w_n u_n \in T \text{ for some } u_0, u_1, \ldots, u_n \,\},\\
\mathrm{Sub}^{\leq r}(T) &= \bigcup_{n=1}^{r} \mathrm{Sub}^n(T),\\
\mathrm{Con}(T) &= \{\, (u, v) \in \Sigma^* \times \Sigma^* \mid uxv \in T \text{ for some } x \,\}.
\end{aligned}$$

In Algorithm 1, for a production $B_0 \to w_0\, B_1\, w_1 \ldots B_n\, w_n$ to be *valid on* $E \subseteq \Sigma^*$ means

$$[\![B_0]\!]^{L_*} \supseteq w_0\, (E \cap [\![B_1]\!]^{L_*})\, w_1 \ldots (E \cap [\![B_n]\!]^{L_*})\, w_n.$$

By the requirement $[\![B]\!]^{L_*} \subseteq \mathrm{Sub}(L_*)$, a production is valid (in the sense of (2)) if and only if it is valid on $\mathrm{Sub}(L_*)$.

Lemma 5 *If K is in the extended regular closure of $\{\, u\backslash L/v \mid (u, v) \in \Sigma^* \times \Sigma^* \,\}$, then there is an $e \in \mathcal{R}$ such that $[\![e]\!]^L = K$ and $(u, v) \in \mathrm{Con}(L)$ for every query atom $(u, v)^\triangleleft$ occurring in e.*

Algorithm 1: Learner for $\mathbf{EXREG}_r(k)$.

Parameters: Positive integers r, k
Data: A positive presentation $t_1, t_2, \dots$ of $L_* \subseteq \Sigma^*$; membership oracle for L_*;
Result: A sequence of grammars $G_1, G_2, \dots$;

$T_0 := \varnothing$; $E_0 := \varnothing$; $J_0 := \varnothing$; $H_0 := \varnothing$; $G_0 := (\{(\varepsilon,\varepsilon)^\triangleleft\}, \Sigma, \varnothing, (\varepsilon,\varepsilon)^\triangleleft)$;
for $i = 1, 2, \dots$ **do**
  $T_i := T_{i-1} \cup \{t_i\}$; $E_i := \mathrm{Sub}(T_i)$;
  if $T_i \not\subseteq L(G_{i-1})$ **then**
    $J_i := \mathrm{Con}(T_i)$; $H_i := \mathrm{Sub}^{\leq r+1}(T_i)$;
  else
    $J_i := J_{i-1}$; $H_i := H_{i-1}$;
  end
  output $G_i := (N_i, \Sigma, P_i, (\varepsilon,\varepsilon)^\triangleleft)$ where
  $N_i := \{\, e \in \mathcal{R} \mid e$ is guarded, every query atom occurring in e is in $\{\, (u,v)^\triangleleft \mid (u,v) \in J_i \,\}$,
    and $\mathrm{size}(e) \leq k \,\}$;
  $P_i := \{\, B_0 \to w_0\, B_1\, w_1 \dots B_n\, w_n \mid (w_0, w_1, \dots, w_n) \in H_i$,
    $B_0, B_1, \dots, B_n \in N_i, B_0 \to w_0\, B_1\, w_1 \dots B_n\, w_n$ is valid on $E_i \,\}$;
end

Proof If $(u,v) \notin \mathrm{Con}(L)$, then $[\![(u,v)^\triangleleft]\!]^L = u\backslash L / v = \varnothing$, so $(u,v)^\triangleleft$ can be replaced with $\varnothing$. ■

Theorem 6 *If $L_* \subseteq \Sigma^*$ is in $\mathbf{EXREG}_r(k)$, then Algorithm 1 converges to a grammar $G = (N, \Sigma, P, S)$ for L_*. Moreover, $([\![B]\!]^{L_*})_{B \in N}$ is an SPP of G.*

Proof Suppose that $G_* = (N_*, \Sigma, P_*, S)$ is a grammar for L_* such that

- the right-hand side of each production in P_* has at most r nonterminals, and
- there is a guarded $e_B \in \mathcal{R}$ with $\mathit{size}(e_B) \leq k$ for each $B \in N_*$ such that $([\![e_B]\!]^L)_{B \in N_*}$ is an SPP of G_*.

We may safely assume that for every nonterminal B of G_*, B is reachable and $L_G(B) \neq \varnothing$. We can then assume that all query atoms $(u,v)^\triangleleft$ in e_B satisfy $(u,v) \in \mathrm{Con}(L_*)$.[6] Finally, we may also safely assume that $e_S = (\varepsilon,\varepsilon)^\triangleleft$.

Let

$$J = \{\, (u,v) \in \Sigma^* \times \Sigma^* \mid (u,v)^\triangleleft \text{ occurs in } e_B \text{ for some } B \in N_* \,\},$$
$$H = \{\, (w_0, w_1, \dots, w_n) \mid B_0 \to w_0\, B_1\, w_1 \dots B_n\, w_n \in P_* \,\}.$$

We have $J \subseteq \mathrm{Con}(L_*)$ and $H \subseteq \mathrm{Sub}^{\leq r+1}(L_*)$. Since $L_* = \bigcup_i T_i$, there is an l such that $J \subseteq \mathrm{Con}(T_l)$ and $H \subseteq \mathrm{Sub}^{\leq r+1}(T_l)$. We distinguish two cases.

6. This is so because applying the procedure in the proof of Lemma 5, we can eliminate all query atoms $(u,v)^\triangleleft$ from e_B such that $(u,v) \notin \mathrm{Con}(L_*)$. Some of the guards of e_B may become $\varnothing$ by this procedure, but at least one guard remains, by the proof of Lemma 3. We can then use simple identities involving $\varnothing$ ($\varnothing \cap e = \varnothing$, $\varnothing \cup e = e$) to turn the resulting expression into a guarded one.

Case 1. $T_i \subseteq L(G_{i-1})$ for all $i \geq l$. In this case, for all $i \geq l$, we have $J_i = J_{i-1}$, $H_i = H_{i-1}$, and $N_i = N_{i-1}$. Since $E_i \supseteq E_{i-1}$ we have $P_i \subseteq P_{i-1}$, so $L(G_i) \subseteq L(G_{i-1})$. It follows that $L_* \subseteq L(G_i)$ for all $i \geq l-1$. Since P_i eventually stabilizes, G_i also stabilizes. When that happens, all productions of G_i will be valid, so the denotations of the nonterminals form a pre-fixed point of G_i. So $L(G_i) \subseteq [\![(\varepsilon,\varepsilon)^\triangleleft]\!]^{L_*} = L_*$. It follows that $L(G_i) = L_*$ and the denotations of the nonterminals of G_i form an SPP of G_i.

Case 2. $T_i \not\subseteq L(G_{i-1})$ for some $i \geq l$. Let m be the least such i. Then $J_m = \mathrm{Con}(T_m) \supseteq \mathrm{Con}(T_l) \supseteq J$, and it follows that $e_B \in N_m$ for each $B \in N_*$. Since $([\![e_B]\!]^{L_*})_{B\in N_*}$ is an SPP of G_*, for each production $B_0 \to w_0\, B_1\, w_1 \ldots B_n\, w_n \in P_*$, the production

$$e_{B_0} \to w_0\, e_{B_1}\, w_1 \ldots e_{B_n}\, w_n$$

is valid. Since $H_m = \mathrm{Sub}^{\leq r+1}(T_m) \supseteq \mathrm{Sub}^{\leq r+1}(T_l) \supseteq H$, all these production are in P_m. It follows that G_m contains a "homomorphic image" of G_*, and so $L(G_m) \supseteq L_*$. It is easy to see that the same is true of all $i \geq m$. By the same reasoning as in Case 1, G_i eventually stabilizes to a grammar for L_* and the denotations of the nonterminals will form an SPP. ■

Theorem 7 *The update time of Algorithm 1 is polynomial in the total lengths of the strings in* T_i.

Proof We give a very rough sketch. In Polish notation, the expressions in N_i are just strings of length $\leq k$ over a certain finite set including $\{\,(u,v)^\triangleleft \mid (u,v) \in J_i\,\}$, so the description size of N_i is polynomial in the total lengths of the strings in T_i. The learner can store the results of the queries "$uyv \in L_*$?" for all $(u,v) \in J_i$ and $y \in E_i$ in a table, and use it to compute $E_i \cap [\![e]\!]^{L_*}$ for each $e \in N_i$ in polynomial time. These sets can then be used to test each candidate production for validity on E_i in polynomial time. (The degree of the polynomial depends on the parameters k and r.) ■

4. Separation

For $x \in \Sigma^*$ and $c \in \Sigma$, let $|x|_c$ denote the number of occurrences of c in x. Kanazawa and Yoshinaka (2021) showed that the language[7] $\overline{O_1} = \{\, x \in \{a,b\}^* \mid |x|_a \neq |x|_b \,\}$ does not have a grammar with the Boolean closure property, so it cannot be targeted by the learning algorithm in that paper. However, this language can be targeted by Algorithm 1 of the present paper with an appropriate choice of r, k.

Theorem 8 *The language* $\overline{O_1}$ *belongs to* **EXREG**.

Proof Consider the following CFG:

$$S \to aT_a \mid aU_abS \mid bT_b \mid bU_baS\,,$$
$$T_a \to \varepsilon \mid aT_a \mid aU_abT_a\,, \qquad U_a \to \varepsilon \mid aU_abU_a\,,$$
$$T_b \to \varepsilon \mid bT_b \mid bU_baT_b\,. \qquad U_b \to \varepsilon \mid bU_baU_b\,,$$

7. Kanazawa and Yoshinaka (2021) erroneously called this language $\overline{O_2}$. What is usually called O_2 is the language $\{\, x \in \{a_1,b_1,a_2,b_2\}^* \mid |x|_{a_1} = |x|_{b_1}$ and $|x|_{a_2} = |x|_{b_2}\,\}$.

Writing B for $L_G(B)$, we can show

$$\begin{aligned}
S &= \overline{O_1},\\
T_a &= \{\, w \in \{a,b\}^* \mid \text{for every prefix } v \text{ of } w,\ |v|_a \geq |v|_b \,\},\\
T_b &= \{\, w \in \{a,b\}^* \mid \text{for every prefix } v \text{ of } w,\ |v|_a \leq |v|_b \,\},\\
U_a &= \{\, w \in \{a,b\}^* \mid |w|_a = |w|_b \text{ and for every prefix } v \text{ of } w,\ |v|_a \geq |v|_b \,\},\\
U_b &= \{\, w \in \{a,b\}^* \mid |w|_a = |w|_b \text{ and for every prefix } v \text{ of } w,\ |v|_a \leq |v|_b \,\}.
\end{aligned}$$

Observe

$$\begin{aligned}
T_a &= \overline{\overline{S}\{b\}\{a,b\}^*}, \qquad & U_a &= \overline{S} \cap T_a,\\
T_b &= \overline{\overline{S}\{a\}\{a,b\}^*}, \qquad & U_b &= \overline{S} \cap T_b.
\end{aligned}$$

This shows that $\overline{O_1} \in \mathbf{EXREG}_2(15)$. (Note that for the nonterminals other than S, we need to add a query atom as a guard.) ■

We give another example:

Theorem 9 *The language $L = S_1 \cup S_2$, where*

$$S_1 = \{\, a^m b^n \mid m \geq n \geq 1 \,\}, \qquad S_2 = \{\, a^m b^n \mid m \geq 1, 2m \leq n \leq 3m \,\},$$

does not have a grammar with the Boolean closure property, but belongs to **EXREG**.

Proof We only give a rough sketch of the proof that L does not have a grammar with the Boolean closure property. (The proof uses a similar idea to the one used in the proof of Theorem 5 of Kanazawa and Yoshinaka (2021).) By applying the pumping lemma to the string of $a^{2p}b^{5p}$ for some sufficiently large p, we can show that any grammar G for L must have a nonterminal A such that

$$\begin{aligned}
S &\Rightarrow_G^* a^{m_1+m_2 i} A b^{n_1+n_2 i} && \text{for all } i \geq 0,\\
A &\Rightarrow_G^+ a^{m_3 i+m_4} b^{n_3 i+n_4} && \text{for all } i \geq 0
\end{aligned}$$

for some $m_1, m_2, m_3, m_4, n_1, n_2, n_3, n_4$ with $m_2 \geq 1$, $2m_2 \leq n_2 \leq 3m_2$, $m_3 \geq 2$, $2m_3 < n_3 < 3m_3$. Suppose that X_A is the A-component of an SPP of G. By the first of the above properties, $a^{2i}b^i \notin X_A$ for all sufficiently large i. By the second property, $a^{m_3 i+m_4} b^{n_3 i+n_4} \in X_A$ for all $i \geq 0$. But for any $(u, v) \in \{a,b\}^* \times \{a,b\}^*$, both $a^{2i}b^i \in u\backslash S_1/v$ and $a^{m_3 i+m_4} b^{n_3 i+n_4} \in u\backslash S_2/v$ must hold for all sufficiently large i. So if X_A is a Boolean combination of sets of the form $u\backslash L/v$, then $a^{2i}b^i \in X_A$ if and only if $a^{m_3 i+m_4} b^{n_3 i+n_4} \in X_A$ for all sufficiently large i, which is clearly a contradiction.

Consider the following grammar for L:

$$S \to S_1 \mid S_2, \quad S_1 \to T \mid aS_1, \quad S_2 \to U \mid abbb \mid aS_2bbb \quad T \to ab \mid aTb, \quad U \to abb \mid aUbb.$$

We have

$$\begin{aligned}
T &= \{\, a^n b^n \mid n \geq 1 \,\} & S_1 &= \{\, a^m b^n \mid m \geq n \geq 1 \,\}\\
&= L \cap \overline{a\backslash L/bb}, & &= a^* T,\\
U &= \{\, a^m b^{2m} \mid m \geq 1 \,\} & S_2 &= \{\, a^m b^n \mid m \geq 1, 2m \leq n \leq 3m \,\}\\
&= L \cap \overline{a\backslash L/b} & &= L \cap Ub^*,\\
& & S &= L.
\end{aligned}$$

This shows that L is in **EXREG**. ■

Theorem 10 *The language*

$$L = \{\, a^l b^m a^n b^q \mid l, m, n, q > 0 \text{ and } l = n \vee m > q \,\}$$

is not in **EXREG**.

Proof Suppose that there is a CFG $G = (N, \{a, b\}, P, S)$ for L that has an SPP $(X_B)_{B \in N}$ consisting of sets in the extended regular closure of the quotients of L. Then there is a finite set $C \subseteq \{a, b\}^* \times \{a, b\}^*$ such that for every nonterminal B, X_B is in the extended regular closure of $\{\, u\backslash L/v \mid (u, v) \in C \,\}$. Let $m = 1 + \max\{\, |uv| \mid (u, v) \in C \,\}$.

Let p be the "Ogden number" for G. We observe that we can apply Ogden's lemma (Ogden, 1968) to the string

$$a^p b^m \boxed{a^p} b^{m+1},$$

where the boxed substring indicates the distinguished positions, and obtain a nonterminal A such that

$$S \Rightarrow_G^* a^{l_1} A a^{l_2} b^{m+1}, \quad A \Rightarrow_G^+ a^s A a^s, \quad A \Rightarrow a^{l_3} b^m a^{l_4},$$

where $s > 0$ and $l_1 + s + l_3 = l_4 + s + l_2 = p$. Then

$$\{\, a^{ns+l_3} b^m a^{l_4+ns} \mid n \geq 0 \,\} \subseteq L_G(A) \subseteq X_A,$$

whereas

$$\{\, a^{n_1 s+l_3} b^m a^{l_4+n_2 s} \mid n_1 \neq n_2 \,\} \cap X_A = \varnothing.$$

It follows that

$$A' = X_A \cap (a^s)^* a^{l_3} b^m a^{l_4} (a^s)^*$$

is not regular. Let

$$R = a^* \cup a^+\{\, b^i \mid i \leq m \,\} \cup \{\, b^i \mid i \leq m \,\} \cup a^+ b^m a^+ \cup \{\, b^i \mid i \leq m \,\} a^+.$$

Then R is closed under substring and $A' \subseteq R$. By Lemma 1, $A' = A' \cap R$ is in the extended regular closure of $\{\, (u\backslash L/v) \cap R \mid (u, v) \in C \,\}$.

To derive a contradiction, it suffices to show that for each $(u, v) \in C$, the following sets are all regular:

(a) $(u\backslash L/v) \cap a^*$

(b) $(u\backslash L/v) \cap a^+\{\, b^i \mid i \leq m \,\}$

(c) $(u\backslash L/v) \cap \{\, b^i \mid i \leq m \,\}$

(d) $(u\backslash L/v) \cap a^+ b^m a^+$

(e) $(u\backslash L/v) \cap \{\, b^i \mid i \leq m \,\} a^+$

This is trivial for (a) and (c), since every context-free language over a one-letter alphabet is regular. (b) and (e) are almost as easy. It remains to consider (d). We distinguish two cases.

Case 1. $(u, v) \notin a^* \times a^* b^*$. Then it is easy to see that $(u\backslash L/v) \cap a^+ b^m a^+ = \varnothing$.

Case 2. $u = a^q$, $v = a^r b^t$. Then since $m > t$, $(u\backslash L/v) \cap a^+ b^m a^+ = a^+ b^m a^+$. ■

5. Extended Regular Closure vs. Star-Free Closure

In the description of the SPP in the proof of Theorem 8, Kleene star is not used in an essential way, since $\{a,b\}^* = \overline{\varnothing}$.[8] So the sets in this SPP are all in the *star-free closure* of $\{\overline{O_1}\}$. Star-free closure is like extended regular closure except for the absence of Kleene star in the repertoire of available operations.[9] Likewise, since $a^* = \overline{\overline{\varnothing}b\overline{\varnothing}}$ and $b^* = \overline{\overline{\varnothing}a\overline{\varnothing}}$ when $\Sigma = \{a,b\}$, the components of the SPP in the proof of Thorem 9 are in the star-free closure of the language L of this theorem.

An obvious question to ask is whether there is a context-free language in **EXREG** that does not have a grammar with the star-free closure property. We have been unable to find such an example, but the following example may be instructive. Consider

$$L = \{\, a^n b^m c^l \mid (n \text{ is odd} \wedge n > m) \vee (n \text{ is even} \wedge n > l) \,\}.$$

L is generated by the following CFG:

$$\begin{aligned} &S \to S_1 \mid S_2, \quad S_1 \to T \mid S_1 c, && S_2 \to aaB \mid aaBc \mid aaS_2 \mid aaS_2 c \mid aaS_2 cc,\\ &\qquad\qquad T \to a \mid aaT \mid aaTb \mid aaTbb, && B \to \varepsilon \mid Bb. \end{aligned}$$

We have

$$\begin{aligned} T &= \{\, a^n b^m \mid n \text{ is odd} \wedge n > m \,\} & B &= b^*,\\ &= (aa)^* ab^* \cap L, & S_2 &= \{\, a^n b^m c^l \mid n \text{ is even} \wedge n > l \,\}\\ S_1 &= \{\, a^n b^m c^l \mid n \text{ is odd} \wedge n > m \,\} & &= (aa)^* b^* c^* \cap L.\\ &= Tc^*, \end{aligned}$$

This shows $L \in \mathbf{EXREG}$.

Note that the use of Kleene star in the above description of T, S_1, S_2 is essential, since $(aa)^*$ is not a star-free regular language. We can show that the sets T, S_1, S_2 are not in the star-free closure of the quotients of L using an analogue of Lemma 1 for the star-free sets and star-free closure. Moreover, the S_1-component of every SPP for this grammar must be S_1, so this grammar does not have the star-free closure property. But this of course does not rule out the possibility that L has a less obvious grammar that does have the star-free closure property.

For instance, here is a rather different grammar for the same language:

$$\begin{aligned} S &\to S_0 \mid S_1 \mid S_2,\\ S_0 &\to aU \mid aS_0 \mid aS_0 c,\\ S_1 &\to T_1 c \mid T_2 cc \mid S_1 c \mid aaS_1 cc,\\ S_2 &\to aaVbb \mid aaVbc \mid aaVcc \mid aaS_2 cc,\\ U &\to \varepsilon \mid aUb, \end{aligned}$$

8. This rewriting puts $\overline{O_1}$ in $\mathbf{EXREG}_2(13)$.
9. The term *star-free closure* is used in this sense by Place and Zeitoun (2019). The study of the *star-free sets*, regular languages that can be described by extended regular expressions without the Kleene star, has a long history. See, e.g., Lawson (2004).

$$T_1 \to a \mid aaT_1bb,$$
$$T_2 \to aaab \mid aaT_2bb,$$
$$V \to \varepsilon \mid Vb \mid aaVbb.$$

Let

$$E = \{\, a^n b^m \mid n \text{ is even and } n = m \,\} = a^*b^* \cap a\backslash L \cap \overline{a\backslash L/b}.$$

We have

$$\begin{aligned}
U &= \{\, a^n b^m \mid n = m \,\} \\
&= E \cup aEb, \\
T_1 &= \{\, a^m b^n \mid m \text{ is odd} \wedge m = n+1 \,\} \\
&= a^*b^* \cap L \cap \overline{L/b}, \\
T_2 &= \{\, a^m b^n \mid m \text{ is odd} \wedge m = n+2 \,\} \\
&= a^*b^* \cap L \cap L/b \cap \overline{L/bb}, \\
V &= \{\, a^m b^n \mid m \text{ is even} \wedge m \le n \,\} \\
&= a^*b^* \cap \overline{a\backslash L/b}, \\
S_0 &= \{\, a^n b^m c^l \mid n > m + l \,\} \\
&\subseteq \{\, a^m b^n c^l \mid m > \max(n, l) \,\} \\
&= L \cap a^+ U c^* \cap (a\backslash L/c), \\
S_1 &= \{\, a^n b^m c^l \mid n \text{ is odd} \wedge m < n \le m + l \,\}, \\
S_2 &= \{\, a^n b^m c^l \mid n \text{ is even} \wedge l < n \le m + l \,\}.
\end{aligned}$$

It is not immediately clear whether or not this grammar has the star-free closure property.

6. Discussion

We have shown that the scope of the distributional learning technique can be significantly extended while maintaining the rough outline of the existing distributional learners.

It is clear that the construction of the set of nonterminals N_i in Algorithm 1 is amenable to "optimization" in several obvious ways.[10] Given that the degree of the polynomial in Theorem 7 is very large, however, the details of the learning algorithm, as well as the most natural definition of the parametrized class $\mathbf{EXREG}_r(k)$, is less urgent an issue than the delineation of the class **EXREG** and its star-free counterpart (discussed in Section 5). There is very little we know about them; for instance, we currently do not even have a single example of a CFL outside of **EXREG** that is not inherently ambiguous. We intend to explore these language classes further in a future work.

Acknowledgments

This work was supported by JSPS KAKENHI Grant Numbers 17K00026 and 18K11150.

10. In particular, the use of $\varnothing$ should be prohibited, except perhaps in the context $\overline{\varnothing}$. It might also be reasonable, in light of the proof of Lemma 1, to demand that every subexpression e' (except $\varnothing$) satisfy $E_i \cap [\![e']\!]^{L_*} \neq \varnothing$. This would be in the same spirit as the requirement that $(u, v) \in J_i$ for each query atom $(u, v)^\triangleleft$.

References

Alexander Clark and Ryo Yoshinaka. Distributional learning of context-free and multiple context-free grammars. In Jeffrey Heinz and José M. Sempere, editors, *Topics in Grammatical Inference*, pages 143–172. Springer, Berlin, 2016.

Makoto Kanazawa and Ryo Yoshinaka. The strong, weak, and very weak finite context and kernel properties. In Frank Drewes, Carlos Martín-Vide, and Bianca Truthe, editors, *Language and Automata Theory and Applications*, pages 77–88, Cham, 2017. Springer International Publishing. ISBN 978-3-319-53733-7. doi: 10.1007/978-3-319-53733-7_5.

Makoto Kanazawa and Ryo Yoshinaka. A hierarchy of context-free languages learnable from positive data and membership queries. In Jane Chandlee, Rémi Eyraud, Jeff Heinz, Adam Jardine, and Menno van Zaanen, editors, *Proceedings of the Fifteenth International Conference on Grammatical Inference*, volume 153 of *Proceedings of Machine Learning Research*, pages 18–31. PMLR, 23–27 Aug 2021. URL https://proceedings.mlr.press/v153/kanazawa21a.html.

Richard Ladner, Nancy Lynch, and Alan Selman. Comparison of polynomial-time reducibilities. In *Proceedings of the Sixth Annual ACM Symposium on Theory of Computing*, pages 110–121, 1974. URL https://doi.org/10.1145/800119.803891.

Mark V. Lawson. *Finite Automata*. Chapman and Hall/CRC, 2004. URL https://doi.org/10.1201/9781482285840.

William Ogden. A helpful result for proving inherent ambiguity. *Mathematical Systems Theory*, 2(3):191–194, 1968.

Thomas Place and Marc Zeitoun. On all things star-free. In Christel Baier, Ioannis Chatzigiannakis, Paola Flocchini, and Stefano Leonardi, editors, *46th International Colloquium on Automata, Languages, and Programming (ICALP 2019)*, volume 132 of *Leibniz International Proceedings in Informatics (LIPIcs)*, pages 126:1–126:14, Dagstuhl, Germany, 2019. Schloss Dagstuhl–Leibniz-Zentrum fuer Informatik. ISBN 978-3-95977-109-2. doi: 10.4230/LIPIcs.ICALP.2019.126. URL http://drops.dagstuhl.de/opus/volltexte/2019/10702.