Connectedness: Papers by and for Sarah VanWagenen
UCLA Working Papers in Linguistics 18: 147–159
Carson T. Schütze & Linnaea Stockall (eds.), 2014

Learning Adjuncts*

Meaghan Fowlie

Human languages include adjuncts, which are grammatically optional elements. Some adjuncts can also be repeated indefinitely. I consider four learnable classes of languages and ask whether these classes include optional and repeated elements, and what input a learner requires in order to generalise from finite to indefinite repetition.

Keywords adjunct, adjective, optionality, repetition, learnability, PDFA, 0-reversible, n-gram, substitutable context free

1 Introduction

At the heart of linguistic study is the question of how such a complex system can be learned by people so young and unformed that they cannot even survive on their own. In describing existent human languages we often hope that the phenomena we encounter will provide some insight into this puzzle: perhaps we have found a universal feature, meaning it could be somehow "built in", sidestepping the necessity for children to learn it; or perhaps we have found a clearcut parameter on which human languages may differ, pointing out a specific fact that children might automatically watch for. Language acquisition studies children's language learning directly, while formal language theory can be used to discover what sorts of grammars are required to describe human language. Learnability theory is the study of mathematical models of language acquisition.

This paper will explore learning models applied to two specific phenomena: optionality and repetition. Adjuncts are by definition optional in that although a sentence will have a different meaning without the adjunct, it is still perfectly grammatical, and the meanings of the sentences differ systematically. For example, in (1) we see that the adjective *red* is optional, and (1-a) entails (1-b).

(1) a. My love is like a **red** rose.
 b. My love is like a rose.

Many human languages allow optional repetition of adjuncts, as for example English:

(2) a. My love is like a **red red** rose.
 b. She's **really really really really really** nice.

*This paper is dedicated to Sarah VanWagenen, who was one of the coolest people I have met.

In this paper I will look at several formal learning models and ask the following 2 questions:

1. Are optionality and repetition possible in this class of languages?
2. What kinds of sentences would the learner need to encounter in order to conclude that a given element can be repeated indefinitely? Omitted?

2 Learnability

Mathematically, a learner is a function from an input text to a grammar.

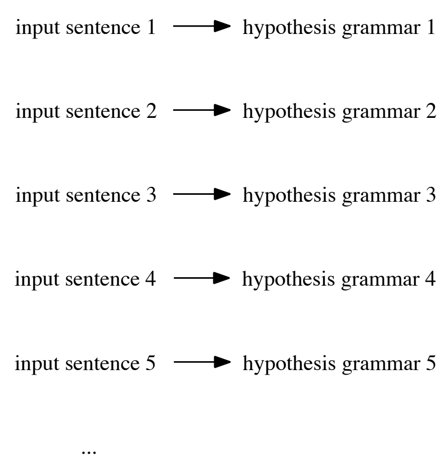


Figure 1: Learning

For example, a very simple learner could simply remember every sentence it's heard. The grammar is then simply the list of sentences. No novel sentences would be generated by this grammar.

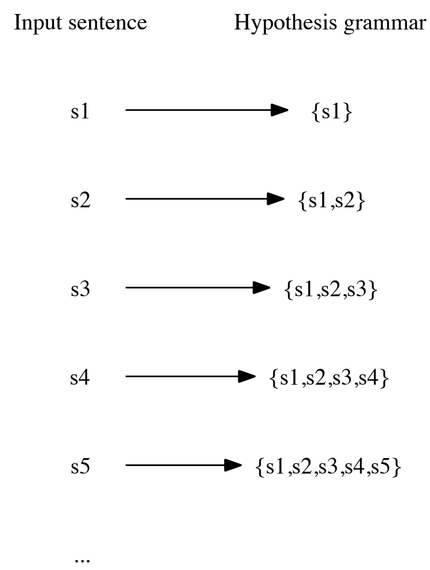


Figure 2: Finite language learner

Such a simple learner cannot learn infinite languages and cannot generalize, so although it is a valid learning model, in that it is a function from a text to a grammar, it is definitely not a model of how babies learn language.

The kinds of patterns in the input that the learner is sensitive to depends on the assumptions that the particular learner makes about the nature of the language.

In current learnability theory, there are two types of learning: Gold learning and PAC learning. They define what it means for a learner to have succeeded in learning a language.

Gold learning, or learning in the limit from positive data, is achieved when the learner eventually converges on exactly the right language. Such learning is very hard; for example there is no one learner that can learn all finite language plus even one infinite language (Gold 1967). However, there are some classes of languages that are known to be Gold-learnable. None of these are human-like languages as of yet.

Probably Approximately Correct (PAC) learning is achieved when the probability that the language is "close enough" to being correct is "high enough". *Close enough* and *high enough* are determined by thresholds set in advance.

Definition 1 (PAC learning)**.** $\forall 0 < \varepsilon < 0.5, 0 < \delta < 0.5$ a Probably Approximately Correct learner outputs hypothesis grammars that are, with probability $1-\delta$, ε-close to correct. (Valiant 1984)

Learners of Regular languages are much better understood than those for languages higher on the Chomsky hierarchy (Chomsky 1959). Since human languages are known to be Mildly Context-Sensitive (Joshi 1985), learning algorithms for such languages are clearly more relevant to actual human language learning; however, as research into such learners is still in its infancy, and since our understanding of Regular learners has driven higher-level learners (see for example Clark, Eyraud, & Habraud Clark et al. (2008)'s substitutable CF learner and Yoshinaka Yoshinaka (2008)'s *k,l*-substitutable CF learner, which are extensions of Angluin (1982)'s 0- and *k*-reversible learners to the context free level), I will look at learners low on the Chomsky hierarchy as well. I provide here four examples.

3 Repetition and Optionality

Here I formalize the notions of optionality and repetition in language L. Note that it only makes sense to define optionality and repetition of a string in a context.

Definition 2 (Language)**.** Given a finite set Σ, Σ^* is the set of all finite sequences of elements of Σ. L is a language iff $L \subseteq \Sigma^*$

Definition 3 (Context)**.** A *context* is a pair (u, v) where $u, v \in \Sigma^*$

Definition 4 (Optional)**.** $x \in \Sigma^*$ is *optional in context (u,v)* iff $uv \in L$ and $uxv \in L$

Definition 5 (Repeatable)**.** $x \in \Sigma^*$ is *repeatable in context (u,v)* iff $ux^+v \subseteq L$, where x^+ means 1 or more *x*s.[2]

It is not obvious what exactly it means to ask whether optionality and repetition are learnable. A simple interpretation of *is repetition learnable?* is *are there any learnable classes $\mathscr{L}$ such that for some $L \in \mathscr{L}, \exists u,x,v \in \Sigma^*$ such that $ux^+v \subseteq L$?* That is, do any learnable classes have even one incidence of repetition in even one language? This question is easily answered: yes. For example, a^* is 0-reversible and is therefore learnable, since there is a learner for the class of 0-reversible languages. (See Section 5).

Slightly more interesting is to ask whether a certain learner can learn repetition. This amounts to asking whether the class of languages it can learn includes some L such that $\exists u,x,v \in \Sigma^*, x \neq \varepsilon$, such that $ux^+v \subseteq L$. This question is also easily answered; it suffices to find an example, and they abound. For example, clearly a learner of finite languages will not be able to learn repetition, as indefinite repetition can only occur in infinite languages. All the language classes I will look at in this paper contain a language with a repeating substring.

Or more interest is to ask what a learner must encounter in order to generalize to indefinite repetition or optionality. For example, is hearing an element repeated once in a context enough? (i.e. if the sample contains *uxv* and *uxxv* does the learner guess a language that includes ux^+v?) What other conclusions will it draw about x?

I will now consider four learners for four languages classes.

---

[2]Languages may also have repeated elements, as for example in (i-a). Notice though that this does not follow our definition of repetition: (i-a) requires *exactly* two *fly*'s. One or three are not grammatical.

(i) a. Why won't the **fly fly** away?
b. *Why won't the **fly** away?
c. *Why won't the **fly fly fly** away?

The difference between (2) above and (i) is that the repeated element (*fly*) in (i) has two-way dependencies, while the repeated elements in (2) (*red* and *really*) only have one-way dependencies. The first *fly* is dependent on the determiner *the*, and *the* is also dependent on it: without *fly*, the phrase becomes ungrammatical. The second *fly* is a verb, and is thus dependent among other things on its subject (*the fly*), and the subject is dependent on it. Conversly, in (2-a), *red* is dependent on *rose*, but *rose* is not dependent on *red*: the sentences is just fine without either *red*. Similarly, in (2-b), *really* is dependent on *nice*, but not vice-versa.

4 N-gram learners

An n-gram learner, for some $n \in \mathbf{N}$, learns languages defined entirely by good substrings of length n. It simply memorises all n-grams it encounters, and accepts/generates strings that contain only n-grams from the list it memorized.

For example, suppose a bigram learner encounters *ac, abc*. Let $\rtimes$ and $\ltimes$ mark word boundaries. The learner generates the following grammar:

	$\rtimes$	a	b	c	$\ltimes$
$\rtimes$		$\checkmark$			
a			$\checkmark$	$\checkmark$	
b				$\checkmark$	
c					$\checkmark$
$\ltimes$					

Table 1: Grammar 1

A word is a valid word of the language if it begins with $\rtimes$, ends with $\ltimes$, and contains only bigrams from this grammar. This grammar does not generalize beyond the input strings: only *ac* and *abc* are valid strings.

Suppose now the learner hears a third string, *abbc*. We update the bigram chart, adding *bb*. (The other bigrams in this string are already present.)

	$\rtimes$	a	b	c	$\ltimes$
$\rtimes$		$\checkmark$			
a			$\checkmark$	$\checkmark$	
b			$\boxed{\checkmark}$	$\checkmark$	
c					$\checkmark$
$\ltimes$					

Table 2: Grammar 2

Now we have grammar that generates an infinite language ab^*c, which is *ac* with zero or more *b*s in the middle. For example, the bigrams of *abbbbbbc* are $\{\rtimes a, ab, bb, bc, c\ltimes\}$, just like those of *abbc*. The learner has generalized to indefinite repeatability from a single repetition.

Generalising to x longer than one symbol, we have Theorem 6 which gives a sufficient sample for learning ux^+v.

Theorem 6. *Let $u,x,v \in \Sigma^*$ and take $n \leq |uxv|$. For an n-gram learner to learn a language containing ux^+v, it suffices for the sample to include* $\{ux^dv | d \in \mathbb{N}, 1 < d < (n/|x|+2)\}$

Proof. Let k be the number of complete x's that can fit in an n-gram.

For any $k \geq 0$, the sample will need $ux^{k+2}v = uxx^kxv$. This is needed so that all n-grams needed to cover any part of the first x and yet contain only xs are present. The n-gram starting at the last symbol of the first x may be longer than $|x| \times k+1$ so another x is needed on the other side of the k xs.

$$\begin{array}{ccccc} u_1u_2...u_m & x_1x_2\ldots x_{|x|} & x^k & x_1x_2x_3\ldots x_{|x|} & v_1v_2\ldots v_t \\ & 1 & 2\ldots k|x|-1 & k|x|\ldots n & \end{array}$$

Notice that for $k = 0$ this means we have $uxxv$ to give us the n-grams in the transition from x to x, without which repetition is not possible.

For $k \geq 1$ the sample will also need to include $uxv, ..., ux^kv$ so that the n-grams that include any part of u and v and each possible number of x's that can fit in the n-gram are learned. Visually, for $i < k$:

$$\begin{array}{ccc} u_1u_2\ldots u_m & x^i & v_1v_2\ldots v_t \\ 1 & 2\ldots i|x|-1 & i|x|\ldots n \end{array}$$

□

For the learner to learn ux^*v, i.e. as above but including uv, the sample need merely also include uv.

5 0-reversible learner

A 0-reversible language is a regular language with the property that any two prefixes of a valid sentence that share one suffix share all suffixes. A learner for a 0-reversible language builds a grammar based on this assumption. The procedure is simple: we start with a prefix tree of the input text. First final states are merged. Then we work from the end of the automaton, merging states that share a suffix.

Definition 7 (0-reversible language)**.** $L \in \mathscr{L}_{0\text{-rev}}$ iff $\forall s,t,u,v \in \Sigma^*$ if $su, sv, tu \in L$ then $tv \in L$

Angluin (1982)'s 0-reversible learner generalizes directly from optionality to indefinite repeatability. The learner starts with a prefix tree (Figure 3) and then merges states with any suffix in common. For example, suppose we have two input strings *ac* and *abc*. First the final states are merged since they share the suffix ε (the empty string) (Figure 4). Next, states 1 and 2 are merged, forming a loop, since the prefixes *a* and *ab* share the suffix *c* (Figure 5).

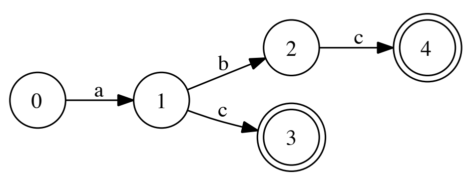


Figure 3: Prefix tree

A relationship between arbitrarily repeatable elements and optional elements is arguably desirable in general. Another way of saying that an element may or may not occur in a situation (in a context or after a state) is to say that the situation is the same whether the element has occurred or not. If this is a situation in which element A may occur, and if the situation is the same once A has occurred, then A may occur again, leaving us in the same

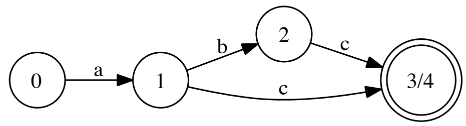


Figure 4: Grammar after states 3 and 4 are merged

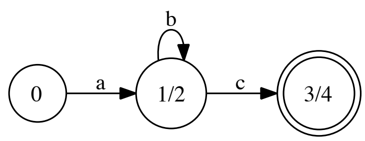


Figure 5: Grammar after states 1 and 2 are merged

situation. For example, the X-bar rule N'→(A) N' means that whether or not A occurs to the left of this N', the result is another N'. This allows indefinite repetition of A. Similarly, in Figure 4, once we get to state 1/2 we remain in state 1/2 no matter how many times *b* occurs, until *c* occurs.

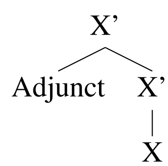


Figure 6: X-bar adjuncts

In 0-reversible languages, optionality and repetition co-occur. $x \in \Sigma^*$ is optional in context C if and only if it is repeatable in C. A pair of simple inductive proofs is enough to show this.

Lemma 8 (0-reversible language: Optionality $\therefore$ Repetition)**.** *Let* $u, v, x \in \Sigma^*$ *and let* $uv, uxv \in L \in \mathscr{L}_{0-rev}$. *Then* $ux^*v \subseteq L$

Proof by induction on number of x's. Base case: ux and u share suffix v.
u also has suffix xv
$\therefore$ ux also has suffix xv,
$\therefore$ $uxxv \in L$.

Inductive step: Suppose $ux^kv, ux^{k+1}v \in L$. Then ux^{k+1} and ux^k share suffix v. ux^k also has suffix xv, so so does ux^{k+1}. Then $ux^{k+2}v \in L$ □

Lemma 9 (0-reversible language: Repetition $\therefore$ Optionality)**.** *Let $ux^kv, ux^{k+1} \in L$ for some $k \geq 0$. Then $uv \in L$.*

Proof by induction on number of x's. **Base case:** Suppose $k = 0$. Then $ux^0v = uv \in L$
Inductive step: suppose $ux^kv, ux^{k+1}v \in L, k > 0$
u^{k-1}, ux^k share suffix xv
ux^k also has suffix v
$\therefore$ so does ux^{k-1}
$\therefore ux^{k-1}v \in L$

□

Theorem 10. *Let L be a 0-reversible language over Σ. Then $\forall k > 0, \forall u, v, x \in \Sigma^*$ we have $(ux^kv \in L \wedge ux^{k+1}v \in L) \Longleftrightarrow (uv \in L \wedge uxv \in L)$*

Proof. By lemmas 8 and 9. □

This means that any learner for a 0-reversible language given a sample that contains ux^kv and ux^{k+1v} **for some** $k \geq 0$ will correctly hypothesize a grammar that generates ux^kv **for all** $k \geq 0$.

6 Substitutable context free languages

Substitutable CF languages (Clark 2010) are the context-free equivalent of 0-reversible languages. 0-reversible languages are defined by common suffix generalization: if two prefixes share one suffix, they share all suffixes. Substitutable CF languages are defined by common *context* generalisation: if two substrings share one context, they share all contexts. This is a learnable class. Briefly, the learner tries all partitions of each input sentence and hypothesizes CF rules of the forms $A \to b$ and $A \to BC$ where A, B, C are sets of contexts and $b \in \Sigma$. In the sample, the right hand sides of the rules have appeared in at least one of the contexts on the left hand sides of the rules.

Like with 0-reversible languages, optionality and repetition co-occur.

Lemma 11 (Substitutable CF: Optionality $\to$ Repetition)**.** *Let $u, v, x \in \Sigma^*$ and $uv, uxv \in L \in \mathscr{L}_{subCF}$ Then $ux^*v \subseteq L(G_i)$.*

Proof. By induction on the number of xs.
Base case: u, ux share context (ε, v).
u also has context (ε, xv) so ux also must have this context.
Therefore $uxxv \in L(G_i)$.
Inductive step: Suppose $ux^kv, ux^{k+1}v \in L$ for some $k \geq 0$. Then ux^{k+1}, ux^k share context (ε, v).
ux^k also has context (ε, xv) so $u^{k+1}x$ also must have this context.
Therefore $ux^{k+2}v \in L(G_i)$.

□

Lemma 12 (Repetition $\to$ Optionality)**.** *Let $u, v, x \in \Sigma^*$ and $ux^nv, ux^{n+1}v \in T[i]$ for some $n \geq 0$ Then $uv \subseteq L(G_i)$.*

Proof. By induction on the number of xs.
Base case: Let $n = 0$. Then $ux^0v = uv \in L$.

Inductive step: Let $n > 0$ and suppose $ux^nv, ux^{n+1}v \in L$. Then ux^{n-1}, ux^n share context (ε, xv). ux^n also has context (ε, v) so ux^{n-1} also must have this context. Therefore $ux^{n-1}v \in L(G_i)$ for all $n \geq 0$.

□

Theorem 13 (Substitutable CF: Optionality $\leftrightarrow$ Repetition)**.** *Let L be a substitutable context free language over Σ. Then $\forall k > 0, \forall u, v, x \in \Sigma^*$ we have $(ux^kv \in L \wedge ux^{k+1}v \in L) \Longleftrightarrow (uv \in L \wedge uxv \in L)$*

Proof. By lemmas 11 and 12. □

This means that any learner for a substitutable CF language given a sample that contains ux^kv and $ux^{k+1}v$ **for some** $k \geq 0$ will correctly hypothesise a grammar that generates ux^kv **for all** $k \geq 0$.

7 Clark & Thollard

Clark & Thollard (2004) describe a PAC (Probably Approximately Correct) learner of probabilistic finite state languages. The learner is similar to Angluin's 0-reversible learner, except that the criterion for merging states is stricter: the similarity of the suffix sets of two states must be within a pre-determined margin for them to be merged. One suffix in common is not enough.

This learner can learn repeatability if the input is representative of the probabilities in the generating grammar; i.e. under normal cicumstances. Unlike the 0-reversible and substitutable CF learners, there is no "short cut".

For example, suppose the language to be learned is ab^*c, with the distribution $p(ab^nc) = (1/2)^{n+1}$. This PDFA generates L:

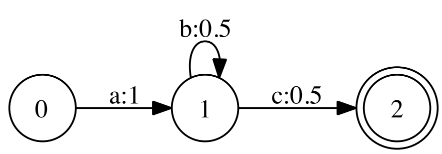
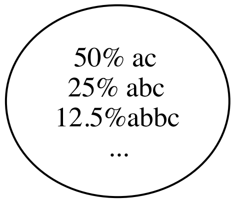
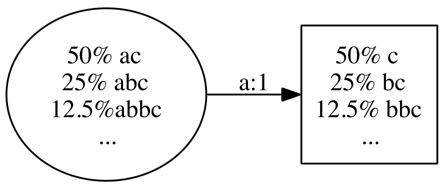


Figure 7: $L = ab^*c$

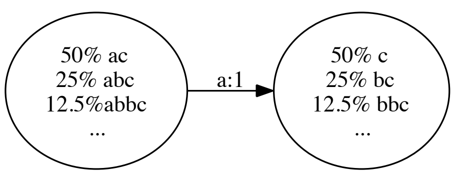
The learner hypothesizes a DFA such that the states are the suffix sets in the sample. It then considers possible "candidate nodes" to follow each node in the hypothesis grammar. If a candidate node is similar enough to an existing node, they are merged (i.e. the existing node is given a new transition). After the first iteration of the learner trained on L, the hypothesis grammar will look like this:



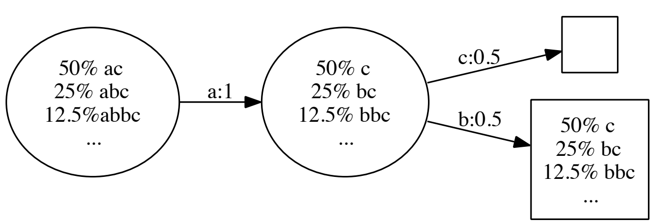
Next candidate nodes are considered. Since all strings start with *a* only one candidate is proposed (shown as a square):



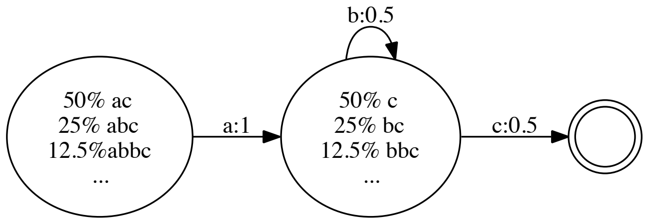
Since these two nodes do not have similar suffix sets, they are not merged, and the candidate becomes a real node:



Next, more samples are drawn and candidates are proposed, one following *b* and the other following *c*. Nothing follows *c* so the latter suffix set is empty.



Since the candidate node with transition *b* and the real node before it have similar suffix sets, the two are merged, yielding a loop.



Clark & Thollard's learner can learn any PDFA with an upper bound on the expected length of strings and size of machine, along with μ-distinguishability, which means that any pair of states is such that there is at least one suffix on which they differ by at least the threshold μ.

Definition 14. For $\mu > 0$ two states $q_1.q_2$ are μ-distinguishable if there is a string s such that the differences in the probabilities of s as a suffix of q_1 and s as a suffix of q_2 is at least μ.

Unlike for n-gram learners, 0-reversible learners, and substitutable CF learners, this learner needs a representative sample to learn repetition: there are no short cuts. Repetition is, however, perfectly learnable.

8 Conclusion

Optionality and repeatability are closely linked both conceptually–repetition is a form of optionality–and in natural language– adjuncts tend to be optional and repeatable. This paper surveyed some basic formal learners and found all to be capable of learning repetition and optionality, and for 0-reversible and substitutible CF learners, optionality and repetition in a context always co-occur.

None of these language classes suffice to describe human syntax. However, substitutable CF languages are not a bad place to start. A major reason to conclude that two phrases have the same category is if they are intersubstitutable. For example, here *the man they call Jayne* and *he* are of the same category and are intersubstitutable. The fact that they are intersubstitutable in the context (ε, *stole money*) is usable as evidence that they are also intersubstitutable in the context (ε, *shot Mal*).

(3) a. **The man they call Jayne** stole money.
b. **He** stole money.
c. **The man they call Jayne** shot Mal.
d. **He** shot Mal.

However, they are not always intersubstitutable, for example in the context (*They built a statue of,* ε).

(4) a. They built a statue of **the man they call Jayne**.
b. *They built a statue of **he**.

Surely humans bring more to learning than intersubstitutability: they have access to context and meaning if nothing else. Substitutability is still a very useful tool, and human learners may well use it.

8.1 Further research

Future research on the learnability of adjuncts will look at learners for language classes closer to human language: CFLs with finite context and finite kernal properties (Clark et al. 2008) and k,l-substitutable Multiple Context Free Languages (Yoshinaka 2009). I am also running artificial language learning experiments asking exactly the same questions: what evidence of repetition do human learners require to generalize to indefinite repetition?

References

Angluin, Dana. 1982. Inference of reversible languages. *Journal of the ACM (JACM)* 29:741–765.

Chomsky, Noam. 1959. On certain formal properties of grammars. *Information and control* 2:137–167.

Clark, Alexander. 2010. Learning context free grammars with the syntactic concept lattice. In *Grammatical inference: Theoretical results and applications*, ed. José M. Sempere and Pedro García, volume 6339 of *Lecture Notes in Computer Science*, 38–51. Springer Berlin Heidelberg. URL `http://dx.doi.org/10.1007/978-3-642-15488-1_5`.

Clark, Alexander, Rémi Eyraud, and Amaury Habrard. 2008. A polynomial algorithm for the inference of context free languages. In *Grammatical inference: Algorithms and applications*, 29–42. Springer.

Clark, Alexander, and Franck Thollard. 2004. Pac-learnability of probabilistic deterministic finite state automata. *The Journal of Machine Learning Research* 5:473–497.

Gold, E Mark. 1967. Language identification in the limit. *Information and Control* 10:447 – 474. URL `http://www.sciencedirect.com/science/article/pii/S0019995867911655`.

Joshi, Aravind. 1985. How much context-sensitivity is necessary for characterizing structural descriptions. In *Natural language processing: Theoretical, computational and psychological perspectives*, ed. D. Dowty, L. Karttunen, and A. Zwicky, 206–250. New York: Cambridge University Press.

Valiant, L. G. 1984. A theory of the learnable. *Commun. ACM* 27:1134–1142. URL `http://doi.acm.org/10.1145/1968.1972`.

Yoshinaka, Ryo. 2008. Identification in the limit of k, l-substitutable context-free languages. In *Grammatical inference: Algorithms and applications*, 266–279. Springer.

Yoshinaka, Ryo. 2009. Learning mildly context-sensitive languages with multidimensional substitutability from positive data. In *Algorithmic Learning Theory*, 278–292. Springer.

Affiliation

Meaghan Fowlie
Department of Linguistics
UCLA
mfowlie@ucla.edu