Learning Languages in the Limit from Positive Information with Finitely Many Memory Changes

Timo Kötzing, Karen Seidel

Hasso-Plattner-Institute, University of Potsdam, Germany

Abstract. We investigate learning collections of languages from texts by an inductive inference machine with access to the current datum and a bounded memory in form of states. Such a bounded memory states (**BMS**) learner is considered successful in case it eventually settles on a correct hypothesis while exploiting only finitely many different states.
We give the complete map of all pairwise relations for an established collection of criteria of successfull learning. Most prominently, we show that non-U-shapedness is not restrictive, while conservativeness and (strong) monotonicity are. Some results carry over from iterative learning by a general lemma showing that, for a wealth of restrictions (the *semantic* restrictions), iterative and bounded memory states learning are equivalent. We also give an example of a non-semantic restriction (strongly non-U-shapedness) where the two settings differ.

1 Introduction

We are interested in the problem of algorithmically learning a description for a formal language (a computably enumerable subset of the set of natural numbers) when presented successively all and only the elements of that language; this is sometimes called *inductive inference*, a branch of (algorithmic) learning theory. For example, a learner M might be presented more and more even numbers. After each new number, M outputs a description for a language as its conjecture. The learner M might decide to output a program for the set of all multiples of 4, as long as all numbers presented are divisible by 4. Later, when M sees an even number not divisible by 4, it might change this guess to a program for the set of all multiples of 2.

Many criteria for deciding whether a learner M is *successful* on a language L have been proposed in the literature. Gold, in his seminal paper [Gol67], gave a first, simple learning criterion, **TxtEx**-*learning*[1], where a learner is *successful* iff, on every *text* for L (listing of all and only the elements of L) it eventually stops changing its conjectures, and its final conjecture is a correct description for the input sequence. Trivially, each single, describable language L has a suitable constant function as an **TxtEx**-learner (this learner constantly outputs a description for L). Thus, we are interested in analyzing for which *classes of languages* $\mathcal{L}$ there is a *single learner* M learning *each* member of $\mathcal{L}$. Sometimes,

[1] **Txt** stands for learning from a *text* of positive examples; **Ex** stands for *explanatory*.

this framework is called *language learning in the limit* and has been studied extensively. For an overview see for example, the textbook [JORS99].

One major criticism of the model suggested by Gold is its excessive use of memory, see for example [CM08]: for each new hypothesis the entire history of past data is available. Iterative learning is the most common variant of learning in the limit which addresses memory constraints: the memory of the learner on past data is just its current hypothesis. Due to the padding lemma [JORS99], this memory is not necessarily void, but only finitely many data can be memorized in the hypothesis. There is a comprehensive body of work on iterative learning, see, e.g., [CK10,CM08,JKMS16,JMZ13,JORS99].

Another way of modelling restricted memory learning is to grant the learner access to not their current hypothesis, but a *state* which can be used in the computation of the next hypothesis (and next state). This was introduced in [CCJS07] and called *bounded memory states (BMS)* learning. It is a reasonable assumption to have a countable reservoir of states. Assuming a computable enumeration of these states, we use natural numbers to refer to them. Note that allowing arbitrary use of all natural numbers as states would effectively allow a learner to store all seen data in the state, thus giving the same mode as Gold's original setting. Probably the minimal way to restrict the use of states is to demand for successful learning that a learner must stop using new states eventually (but may still traverse among the finitely many states produced so far, and may use infinitely many states on data for a non-target language). It was claimed that this setting is equivalent to iterative learning [CCJS07, Remark 38] (this restriction is called *ClassBMS* there, we refer to it by $\mathbf{TxtBMS_*Ex}$). However, this was only remarked for the plain setting of explanatory learning; for further restrictions, the setting is completely unknown, only for explicit constant state bounds a few scattered results are known, see [CCJS07,CK13].

In this paper, we consider a wealth of restrictions, described in detail in Section 2 (after an introduction to the general notation of this paper). Following the approach of giving *maps* of pairwise relations suggested in [KS16], we give a complete map in Figure 1. We note that this map is the same as the map for iterative learning given in [JKMS16], but partially for different reasons.

In Lemma 31 we show that, for many restrictions (the so-called *semantic* restrictions, where only the semantics of hypotheses are restricted) the learning setting with bounded memory states is equivalent to learning iteratively. This proves and generalizes the aforementioned remark in [CCJS07] to a wide class of restrictions.

However, if restrictions are not semantic, then iterative and bounded memory states learning can differ. We show this concretely for *strongly non-U-shaped* learning in Theorem 45. Inspired by cognitive science research [SS82], [MPU$^+$92] a semantic version of this requirement was defined in [BCM$^+$08] and later the syntactic variant was introduced in [CM11]. Both requirements have been extensively studied, see [CC13] for a survey and moreover [CK13], [CK16], [KSS17]. The proof combines the techniques for showing that strong non-U-shapedness restricts iterative learning, as proved in [CK13, Theorem 5.7], and that not every

class strongly monotonically learnable by an iterative learner is strongly non-U-shapedly learnable by an iterative learner, see [JKMS16, Theorem 5]. Moreover, it relies on showing that state decisiveness can be assumed in Lemma 41.

The remainder of Section 4 completes the map given in Figure 1 for the case of syntactic restrictions (since these do not carry over from the setting of iterative learning). All syntactic learning requirements are closely related to strongly locking learners. The fundamental concept of a locking sequence was introduced by [BB75]. For a similar purpose than ours [JKMS16] introduced strongly locking learners. We generalize their construction for certain syntactically restricted iterative learners from a strongly locking iterative learner. Finally, we obtain that all non-semantic learning restrictions also coincide for $\mathbf{BMS}_*$-learning.

2 Learners, Success Criteria and other Terminology

As far as possible, we follow [JORS99] on the learning theoretic side and [Odi99] for computability theory. We recall the most essential notation and definitions.

We let $\mathbb{N}$ denote the *natural numbers* including 0. For a function f we write $\mathrm{dom}(f)$ for its *domain* and $\mathrm{ran}(f)$ for its *range*.

Further, $X^{<\omega}$ denotes the *finite sequences* over the set X and X^{ω} stands for the *countably infinite sequences* over X. For every $\sigma \in X^{<\omega}$ and $t \leqslant |\sigma|$, $t \in \mathbb{N}$, we let $\sigma[t] := \{(s, \sigma(s)) \mid s < t\}$ denote the *restriction of σ to t*. Moreover, for sequences $\sigma, \tau \in X^{<\omega}$ their concatenation is denoted by $\sigma^\frown\tau$. Finally, we write $\mathrm{last}(\sigma)$ for the last element of σ , $\sigma(|\sigma|-1)$, and σ^- for the initial segment of σ without $\mathrm{last}(\sigma)$, i.e. $\sigma[|\sigma|-1]$. Clearly, $\sigma = \sigma^-{}^\frown\mathrm{last}(\sigma)$.

For a finite set $D \subseteq \mathbb{N}$ and a finite sequence $\sigma \in X^{<\omega}$, we denote by $\langle D \rangle$ and $\langle \sigma \rangle$ a canonical index for D or σ, respectively. Further, we fix a Gödel pairing function $\langle ., . \rangle$ with two arguments.

If we deal with (a subset of) a cartesian product or Gödel pairs, we are going to refer to the *projection functions* to the first or second coordinate by pr_1 and pr_2, respectively.

Let $L \subseteq \mathbb{N}$. We interpret every $n \in \mathbb{N}$ as a code for a word. If L is recursively enumerable, we call L a *language*.

We fix a programming system φ as introduced in [RC94]. Briefly, in the φ-system, for a natural number p, we denote by φ_p the partial computable function with program code p. We call p an *index* for W_p defined as $\mathrm{dom}(\varphi_p)$.

In reference to a Blum complexity measure Φ_p, for all $p, t \in \mathbb{N}$, we denote by $W_p^t \subseteq W_p$ the recursive set of all natural numbers less or equal to t, on which the machine executing p halts in at most t steps, i.e. $W_p^t = \{x \mid x \leqslant t \wedge \Phi_p(x) \leqslant t\}$. Moreover, the well-known s-m-n theorem gives finite and infinite recursion theorems, see [Cas94], [Odi99]. We will refer to Case's Operator Recursion Theorem ORT in its 1-1-form, [Cas74].

Throughout the paper, we let $\Sigma = \mathbb{N} \cup \{\#\}$ be the input alphabet with $n \in \mathbb{N}$ interpreted as code for a word in the language and # interpreted as pause symbol, i.e. no new information. Further, let $\Omega = \mathbb{N} \cup \{?\}$ be the output alphabet with $p \in \mathbb{N}$ interpreted as φ-index and ? as no hypothesis or repetition of the last

hypothesis, if existent. A function with range Ω is called a hypothesis generating function.

A *learner* is a (partial) computable function $M : \mathrm{dom}(M) \subseteq \Sigma^{<\omega} \to \Omega$. The set of all total computable functions $M : \Sigma^{<\omega} \to \Omega$ is denoted by $\mathcal{R}$.

Let $f \in \Sigma^{<\omega} \cup \Sigma^{\omega}$, then the *content of* f, defined as $\mathrm{content}(f) := \mathrm{ran}(f) \backslash \{\#\}$, is the set of all natural numbers, about which f gives some positive information. $\mathbf{Txt}(L) := \{T \in \Sigma^{\omega} \mid \mathrm{content}(T) = L\}$ denotes *set of all texts for* L.

Definition 21 *Let M be a learner. M is an* iterative learner *or* **It**-learner*, for short $M \in \mathbf{It}$, if there is a computable (partial) hypothesis generating function $h_M : \Omega \times \Sigma \to \Omega$ such that $M = h_M^{\ddagger}$ where $h_M^{\ddagger}$ is defined on finite sequences by*

$$h_M^{\ddagger}(\epsilon) = ?; \qquad h_M^{\ddagger}(\sigma^\frown x) = h_M(h_M^{\ddagger}(\sigma), x).$$

Definition 22 *Let M be a learner. M is a* bounded memory states learner *or* **BMS**-learner*, for short $M \in \mathbf{BMS}$, if there are a computable (partial) hypothesis generating function $h_M : \mathbb{N} \times \Sigma \to \Omega$ and a computable (partial) state transition function $s_M : \mathbb{N} \times \Sigma \to \mathbb{N}$ such that $\mathrm{dom}(h_M) = \mathrm{dom}(s_M)$ and $M = h_M^*$ where h_M^* and s_M^* are defined on finite sequences by*

$$s_M^*(\epsilon) = 0; \qquad h_M^*(\sigma^\frown x) = h_M(s_M^*(\sigma), x); \qquad s_M^*(\sigma^\frown x) \quad = s_M(s_M^*(\sigma), x).$$

We now clarify what we mean by succesful learning.

Definition 23 *Let M be a learner and $\mathcal{L}$ a collection of languages.*

1. *Let $L \in \mathcal{L}$ be a language and $T \in \mathbf{Txt}(L)$ a text for L presented to M.*
   (a) *We call $h = (h_t)_{t \in \mathbb{N}} \in \Omega^{\omega}$, where $h_t := M(T[t])$ for all $t \in \mathbb{N}$, the* learning sequence of M on T.
   (b) M learns L from T in the limit*, for short M **Ex**-learns L from T or $\mathbf{Ex}(M, T)$, if there exists $t_0 \in \mathbb{N}$ such that $W_{h_{t_0}} = \mathrm{content}(T)$ and $\forall t \geqslant t_0$ $(\, h_t \neq ? \;\Rightarrow\; h_t = h_{t_0} \,)$.*
2. *M* learns *$\mathcal{L}$ in the limit, for short M* **Ex**-learns *$\mathcal{L}$, if $\mathbf{Ex}(M, T)$ for every $L \in \mathcal{L}$ and every $T \in \mathbf{Txt}(L)$.*

Definition 24 *Let $\mathcal{L}$ be a collection of languages. $\mathcal{L}$ is* learnable in the limit *or* **Ex**-learnable*, if there exists a learner M that* **Ex**-learns *$\mathcal{L}$.*

In our investigations, the most important additional requirement on a successful learning process for a **BMS**-learner is to use finitely many states only, as stated in the following definition.

Definition 25 *Let M be a* **BMS**-learner *and $T \in \mathbf{Txt}$. We say that M* uses finitely many memory states on *T, for short $\mathbf{BMS}_*(M, T)$, if $\{\, s_M^*(T[t]) \mid t \in \mathbb{N} \,\}$ is finite.*

Let L be a language. M is said to $\mathbf{BMS}_*\mathbf{Ex}$-learn *$L$, if $\mathbf{BMS}_*\mathbf{Ex}(M, T)$ for every text $T \in \mathbf{Txt}(L)$.*

In [CCJS07, Rem. 38] it is claimed that $\mathbf{BMS}_*$-learners and iterative learners are equally powerful on texts. This also follows from our more general Lemma 31.

We list the most common additional requirements regarding the learning sequence, which may tag a learning process just like $\mathbf{BMS}_*$ above. For this we first recall the notion of consistency of a sequence with a set. For $f \in \Sigma^{<\omega} \cup \Sigma^{\omega}$ and $A \subseteq \Sigma$ we say f *is consistent with* A if and only if $\mathrm{content}(f) \subseteq A$.

The listed properties of the learning sequence have been at the center of different investigations. Studying how they relate to one another did begin in [KP16], [KS16], [JKMS16] and [AKS18].

Definition 26 *Let M be a learner, $T \in \mathbf{Txt}$ and $h = (h_t)_{t\in\mathbb{N}} \in \Omega^{\omega}$ the learning sequence of M on T, i.e. $h_t = M(T[t])$ for all $t \in \mathbb{N}$. We write*

1. $\mathbf{Cons}(T[t], W_{h_t})$*, if* $\{T(s) \mid s < t\}\backslash\{\#\} \subseteq W_{h_t}$.
2. $\mathbf{Cons}(M,T)$ *([Ang80]), if M is* consistent on T*, i.e., for all t holds* $\mathbf{Cons}(T[t], W_{h_t})$.
3. $\mathbf{Conv}(M,T)$ *([Ang80]), if M is* conservative on T*, i.e., for all s,t with $s \leqslant t$ holds* $\mathbf{Cons}(T[t], W_{h_s}) \Rightarrow h_s = h_t$.
4. $\mathbf{Dec}(M,T)$ *([OSW82]), if M is* decisive on T*, i.e., for all r,s,t with $r \leqslant s \leqslant t$ holds $W_{h_r} = W_{h_t} \Rightarrow W_{h_r} = W_{h_s}$.*
5. $\mathbf{Caut}(M,T)$ *([OSW86]), if M is* cautious on T*, i.e., for all s,t with $s \leqslant t$ holds $\neg W_{h_t} \subsetneq W_{h_s}$.*
6. $\mathbf{WMon}(M,T)$ *([Jan91],[Wie91]), if M is* weakly monotonic on T*, i.e., for all s,t with $s \leqslant t$ holds* $\mathbf{Cons}(T[t], W_{h_s}) \Rightarrow W_{h_s} \subseteq W_{h_t}$.
7. $\mathbf{Mon}(M,T)$ *([Jan91],[Wie91]), if M is* monotonic on T*, i.e., for all s,t with $s \leqslant t$ holds $W_{h_s} \cap \mathrm{content}(T) \subseteq W_{h_t} \cap \mathrm{content}(T)$.*
8. $\mathbf{SMon}(M,T)$ *([Jan91],[Wie91]), if M is* strongly monotonic on T*, i.e., for all s,t with $s \leqslant t$ holds $W_{h_s} \subseteq W_{h_t}$.*
9. $\mathbf{NU}(M,T)$ *([BCM+08]), if M is* non-U-shaped on T*, i.e., for all r,s,t with $r \leqslant s \leqslant t$ holds $W_{h_r} = W_{h_t} = \mathrm{content}(T) \Rightarrow W_{h_r} = W_{h_s}$.*
10. $\mathbf{SNU}(M,T)$ *([CM11]), if M is* strongly non-U-shaped on T*, i.e., for all r,s,t with $r \leqslant s \leqslant t$ holds $W_{h_r} = W_{h_t} = \mathrm{content}(T) \Rightarrow h_r = h_s$.*
11. $\mathbf{SDec}(M,T)$ *([KP16]), if M is* strongly decisive on T*, i.e., for all r,s,t with $r \leqslant s \leqslant t$ holds $W_{h_r} = W_{h_t} \Rightarrow h_r = h_s$.*
12. $\mathbf{Wb}(M,T)$ *([KS16]), if M is* witness-based on T*, i.e., for all r,t such that for some s with $r < s \leqslant t$ holds $h_r \neq h_s$ holds* $\mathrm{content}(T[s]) \cap (W_{h_t}\backslash W_{h_r}) \neq \varnothing$.

It is easy to see that $\mathbf{Conv}(M,T)$ implies $\mathbf{SNU}(M,T)$ and $\mathbf{WMon}(M,T)$; $\mathbf{SDec}(M,T)$ implies $\mathbf{Dec}(M,T)$ and $\mathbf{SNU}(M,T)$; $\mathbf{SMon}(M,T)$ implies all of $\mathbf{Caut}(M,T)$, $\mathbf{Dec}(M,T)$, $\mathbf{Mon}(M,T)$, $\mathbf{WMon}(M,T)$ and finally $\mathbf{Dec}(M,T)$, $\mathbf{WMon}(M,T)$ and $\mathbf{SNU}(M,T)$ imply $\mathbf{NU}(M,T)$. Figure 1 includes the resulting backbone with arrows indicating the aforementioned implications. Further, $\mathbf{Wb}(M,T)$ implies $\mathbf{Conv}(M,T)$, $\mathbf{SDec}(M,T)$ and $\mathbf{Caut}(M,T)$.

In order to characterize what successful learning means, these predicates may be combined with the explanatory convergence criterion. For this, we let $\Delta := \{\mathbf{Caut}, \mathbf{Conv}, \mathbf{Dec}, \mathbf{SDec}, \mathbf{WMon}, \mathbf{Mon}, \mathbf{SMon}, \mathbf{NU}, \mathbf{SNU}, \mathbf{T}\}$ denote

the set of *admissible learning restrictions*, with $\mathbf{T}$ standing for no restriction. Further, a *learning success criterion* is a predicate being the intersection of the convergence criterion $\mathbf{Ex}$ with arbitrarily many admissible learning restrictions. This means that the sequence of hypotheses has to converge and in addition has the desired properties. Therefore, the collection of all learning success criteria is $\{\bigcap_{i=0}^{n} \delta_i \cap \mathbf{Ex} \mid n \in \mathbb{N}, \forall i \leqslant n(\delta_i \in \Delta)\}$. Note that plain explanatory convergence is a learning success criterion by letting $n = 0$ and $\delta_0 = \mathbf{T}$.

We refer to all $\delta \in \{\mathbf{Caut}, \mathbf{Cons}, \mathbf{Dec}, \mathbf{Mon}, \mathbf{SMon}, \mathbf{WMon}, \mathbf{NU}, \mathbf{T}\}$ also as *semantic* learning restrictions, as they do not require the learner to settle on exactly one hypothesis. More formally, if texts T_1, T_2 are such that for all $t \in \mathbb{N}$ holds $W_{M(T_1[t])} = W_{M(T_2[t])}$, then $\delta(M, T_1)$ and $\delta(M, T_2)$ are equivalent.

In order to state observations about how two ways of defining learning success relate to each other, the learning power of the different settings is encapsulated in notions $[\alpha\mathbf{Txt}\beta]$. A collection of languages $\mathcal{L}$ is in $[\alpha\mathbf{Txt}\beta]$, if there is a learner with property α that *β-learns $\mathcal{L}$*. We do not use separators in the notation to stay consistent with established notation in the field that was inspired by [JORS99]. Whenever β includes $\mathbf{BMS}_*$ it is understood that we are only considering $\mathbf{BMS}$-learners.

The proofs of Lemmata 31 and 41 employ the following property of learning requirements and learning success criteria, that applies to all such considered in this paper.

Definition 27 *Denote the set of all unbounded and non-decreasing functions by $\mathfrak{S}$, i.e., $\mathfrak{S} := \{\mathfrak{s} : \mathbb{N} \to \mathbb{N} \mid \forall x \in \mathbb{N}\, \exists t \in \mathbb{N}\colon \mathfrak{s}(t) \geqslant x$ and $\forall t \in \mathbb{N}\colon \mathfrak{s}(t+1) \geqslant \mathfrak{s}(t)\}$. Then every $\mathfrak{s} \in \mathfrak{S}$ is a so called* simulating function.

A predicate β on pairs of learners and texts allows for simulation on equivalent text*, if for all simulating functions $\mathfrak{s} \in \mathfrak{S}$, all texts $T, T' \in \mathbf{Txt}$ and all learners M, M' holds: Whenever we have* $\mathrm{content}(T'[t]) = \mathrm{content}(T[\mathfrak{s}(t)])$ *and $M'(T'[t]) = M(T[\mathfrak{s}(t)])$ for all $t \in \mathbb{N}$, from $\beta(M, T)$ we can conclude $\beta(M', T')$.*

Intuitively, as long as the learner M' conjectures $h'_t = h_{\mathfrak{s}(t)} = M(T[\mathfrak{s}(t)])$ at time t and has, in form of $T'[t]$, the same data available as was used by M for this hypothesis, M' on T' is considered to be a simulation of M on T.

It is easy to see that all learning success criteria considered in this paper allow for simulation on equivalent text.

3 Relations between Semantic Learning Requirements

The following lemma formally establishes the equal learning power of iterative and $\mathbf{BMS}_*$-learning for all learning success criteria but **Conv**, **SDec** and **SNU**. We are going to prove in Section 4 that this is not true for these three non-semantic additional requirements.

Lemma 31 *Let δ allow for simulation on equivalent text.*

1. *We have* $[\mathbf{Txt}\mathbf{BMS}_*\delta\mathbf{Ex}] \supseteq [\mathbf{It}\mathbf{Txt}\delta\mathbf{Ex}]$.

2. *If δ is semantic then* $[\mathbf{TxtBMS}_*\delta\mathbf{Ex}] = [\mathbf{ItTxt}\delta\mathbf{Ex}]$.

While 1 and "$\supseteq$" in 2 are easy to verify by using the hypotheses as states, the other inclusion in 2 is more challenging. The iterative learner constructed from the **BMS**-learner M uses the hypotheses of M on an equivalent text and additionally pads a subgraph of the translation diagram of M to it.

With Lemma 31 the following results transfer from learning with iterative learners and it remains to investigate the relations to and between the non-semantic requirements **Conv**, **SDec** and **SNU**.

Theorem 32 1. $[\mathbf{TxtBMS}_*\mathbf{NUEx}] = [\mathbf{TxtBMS}_*\mathbf{Ex}]$
2. $[\mathbf{TxtBMS}_*\mathbf{DecEx}] = [\mathbf{TxtBMS}_*\mathbf{WMonEx}] = [\mathbf{TxtBMS}_*\mathbf{CautEx}] = [\mathbf{TxtBMS}_*\mathbf{Ex}]$
3. $[\mathbf{TxtBMS}_*\mathbf{MonEx}] \subsetneq [\mathbf{TxtBMS}_*\mathbf{Ex}]$
4. $[\mathbf{TxtBMS}_*\mathbf{SMonEx}] \subsetneq [\mathbf{TxtBMS}_*\mathbf{MonEx}]$

Proof. The respective results for iterative learners can be found in [CM08, Theorem 2], [JKMS16, Theorem 10], [JKMS16, Theorem 3] and [JKMS16, Theorem 2]. □

4 Relations to and between Syntactic Learning Requirements

The following lemma establishes that we may assume $\mathbf{BMS}_*$-learners to never go back to withdrawn states. This is essential in almost all of the following proofs. It can also be used to simplify the proof of Lemma 31.

Lemma 41 *Let β be a learning success criterion allowing for simulation on equivalent text and $\mathcal{L} \in [\mathbf{TxtBMS}_*\beta]$. Then there is a* **BMS***-learner N such that N never returns to a withdrawn state and $\mathbf{BMS}_*\beta$-learns $\mathcal{L}$ from texts.*

With the latter result we can show that strongly monotonically $\mathbf{BMS}_*$-learnability does not imply strongly non-U-shapedly $\mathbf{BMS}_*$-learnability.

Theorem 42 $[\mathbf{TxtBMS}_*\mathbf{SMonEx}] \not\subseteq [\mathbf{TxtBMS}_*\mathbf{SNUEx}]$

In the proof a self-learning **BMS**-learner M is defined and with a tailored ORT-argument there can not be a **BMS**-learner strongly non-U-shapedly learning all languages that M learns strongly monotonically.

For inferring the relations between the syntactic learning requirements **SNU**, **SDec** and **Conv**, we refer to **Wb**. All these criteria are closely related to strongly locking learners. The learnability of every language L by a learner M is witnessed by a sequence σ, consistent with L, such that $M(\sigma)$ is an index for L and no extension of σ consistent with L will lead to a mind-change of M. Such a sequence σ is called *(sink-)locking sequence for M on L*. A learner M acts strongly locking on a language L, if for every text T for L there is an initial segment σ of T that is a locking sequence for M on L.

The proof of the following theorem generalizes the construction of a conservative and strongly decisive iterative learner from a strongly locking iterative learner in [JKMS16, Theorem 8]. With it we obtain in the Corollary thereafter, that all non-semantic learning restrictions coincide.

Theorem 43 *Let $\mathcal{L}$ be a set of languages* $\mathbf{BMS_*Ex}$*-learned by a strongly locking* $\mathbf{BMS}$*-learner. Then* $\mathcal{L} \in [\mathbf{TxtBMS_*WbEx}]$.

The construction of the witness-based learner proceeds in two steps. First, we construct a learner $\mathbf{BMS_*}$-learning $\mathcal{L}$ locally conservatively, as defined in [JLZ07], requiring the last datum to violate consistency with the former hypothesis. Second, from the aforementioned locally conservative learner, we obtain a new learner that $\mathbf{BMS_*Ex}$-learns $\mathcal{L}$ in a witness-based fashion. We will do this by keeping track of all data having caused a mind-change so far. More concretely, we alter the text by excluding mind-change data causing another mind-change and make sure that the witness for the mind-change is contained in all future hypotheses.

With the latter theorem it is straightforward to observe that in the $\mathbf{BMS_*Ex}$-setting conservative, strongly decisive and strongly non-U-shaped $\mathbf{Ex}$-learning are equivalent.

Corollary 44 *For all* $\gamma, \delta \in \{\mathbf{Conv}, \mathbf{SDec}, \mathbf{SNU}\}$ *holds* $[\mathbf{TxtBMS_*}\gamma\mathbf{Ex}] = [\mathbf{TxtBMS_*}\delta\mathbf{Ex}]$.

By [JKMS16, Theorem 2] and Lemma 31 we obtain $[\mathbf{TxtBMS_*ConvEx}] \not\subseteq [\mathbf{TxtBMS_*SMonEx}]$. From this we conclude with Theorem 42 and Corollary 44 that $[\mathbf{TxtBMS_*ConvEx}] \perp [\mathbf{TxtBMS_*SMonEx}]$.

Similarly, with [JKMS16, Theorem 3] and Lemma 31 $[\mathbf{TxtBMS_*ConvEx}] \not\subseteq [\mathbf{TxtBMS_*MonEx}]$. As $[\mathbf{TxtBMS_*MonEx}] \not\subseteq [\mathbf{TxtBMS_*SNUEx}]$ by Theorem 42, with Corollary 44 $[\mathbf{TxtBMS_*ConvEx}] \perp [\mathbf{TxtBMS_*MonEx}]$.

Because Theorem 42 also reproves $[\mathbf{TxtBMS_*SNUEx}] \subsetneq [\mathbf{TxtBMS_*Ex}]$, first observed in [CK13, Th. 3.10], we completed the map for $\mathbf{BMS_*Ex}$-learning from texts.

As the relations equal the ones for $\mathbf{It}$-learning, naturally the question arises, whether a result similar to Lemma 31 can be observed for the syntactic learning criteria. In the following we show that this is not the case.

Theorem 45 $[\mathbf{ItTxtSNUEx}] \subsetneq [\mathbf{TxtBMS_*SNUEx}]$

Proof. By Lemma 31 we have $[\mathbf{ItTxtSNUEx}] \subseteq [\mathbf{TxtBMS_*SNUEx}]$.

We consider the $\mathbf{BMS}$-learner M initialized with state $\langle\langle ?, 0\rangle, \langle\varnothing\rangle\rangle$ and h_M and s_M for every $\langle e, \xi\rangle \in \Omega$, $D \subseteq \mathbb{N}$ finite and $x \in \Sigma$ defined by:

$$s_M(\langle\langle e, \xi\rangle, \langle D\rangle\rangle, x) = \begin{cases} \langle\langle e, \xi\rangle, \langle D\rangle\rangle, & \text{if } x \in D \cup \{\#\} \vee \\ & \quad \mathrm{pr}_1(\varphi_x(\langle e, \xi\rangle)\downarrow) = e; \\ \langle\varphi_x(\langle e, \xi\rangle), \langle D \cup \{x\}\rangle\rangle, & \text{else if } \mathrm{pr}_1(\varphi_x(\langle e, \xi\rangle)\downarrow) \neq e; \\ \uparrow, & \text{otherwise.} \end{cases}$$

$$h_M(\langle\langle e,\xi\rangle,\langle D\rangle\rangle,x) = \begin{cases} e, & \text{if } x \in D \cup \{\#\} \ \vee \\ & \quad \mathrm{pr}_1(\,\varphi_x(\langle e,\xi\rangle)\!\downarrow\,) = e; \\ \mathrm{pr}_1(\,\varphi_x(\langle e,\xi\rangle)\,), & \text{else if } \mathrm{pr}_1(\,\varphi_x(\langle e,\xi\rangle)\!\downarrow\,) \neq e; \\ \uparrow, & \text{otherwise.} \end{cases}$$

Additionally to the last hypothesis as well as exactly the data that already lead to a mind-change of M, some parameter ξ is stored, indicating whether a further mind-change may cause a syntactic U-shape.

Let $\mathcal{L} = \mathbf{TxtBMS}_*\mathbf{SNUEx}(M)$. We will show that there is no iterative learner **ItTxtSNUEx**-learning $\mathcal{L}$. Assume N is an iterative learner with hypothesis generating function h_N and $\mathcal{L} \subseteq \mathbf{ItTxtEx}(N)$.

We obtain $L \in \mathcal{L}\backslash\mathbf{ItTxtSNUEx}(N)$ by applying 1-1 ORT [Cas74] referring to the Σ_1-predicates MC and NoMC, expressing that N does (not) perform a mind-change on a text built from parameters $a, b \in \mathcal{R}$. More specifically, the predicates state that N does converge and (not) make a mind-change when observing $\sigma \in \Sigma^{<\omega}$ after having observed $a[i]\frown b(i)\frown\#^{\ell_i}$, with $i \in \mathbb{N}$.

$$\begin{aligned} \psi_i(\ell) &\Leftrightarrow N(a[i]\frown b(i)\frown\#^{\ell}) = N(a[i]\frown b(i)\frown\#^{\ell+1}); \\ \mathrm{NoMC}(i,\sigma) &\Leftrightarrow \exists \ell_i \in \mathbb{N}\,(\,\psi_i(\ell_i) \ \wedge\ \forall \ell < \ell_i\, \neg\psi_i(\ell) \ \wedge \\ &\qquad\qquad N(a[i]\frown b(i)\frown\#^{\ell_i}\frown\sigma)\!\downarrow = N(a[i]\frown b(i)\frown\#^{\ell_i})\,); \\ \mathrm{MC}(i,\sigma) &\Leftrightarrow \exists \ell_i \in \mathbb{N}\,(\,\psi_i(\ell_i) \ \wedge\ \forall \ell < \ell_i\, \neg\psi_i(\ell) \ \wedge \\ &\qquad\qquad N(a[i]\frown b(i)\frown\#^{\ell_i}\frown\sigma)\!\downarrow \neq N(a[i]\frown b(i)\frown\#^{\ell_i})\,). \end{aligned}$$

By 1-1 ORT [Cas74], applied to the recursive operator implicit in the following case distinction, there are recursive total functions a, b, e_1, e_2 with pairwise disjoint ranges and $e_0 \in \mathbb{N}$, such that for all $i, \xi \in \mathbb{N}$, $e \in \Omega$

$$\varphi_{a(i)}(\langle e,\xi\rangle) = \begin{cases} \langle e_0,\xi\rangle, & \text{if } e \in \{?, e_0\}; \\ \langle e_1(k),1\rangle, & \text{else if } \xi = 0, i \text{ even and } \exists k \leqslant i\,(\,e = e_1(k)\,); \\ \langle e_1(k),2\rangle, & \text{else if } \xi = 0, i \text{ odd and } \exists k \leqslant i\,(\,e = e_1(k)\,); \\ \langle e_2(k),0\rangle, & \text{else if } \xi = 1, i \text{ odd and } \exists k \leqslant i\,(\,e = e_1(k)\,); \\ \langle e_2(k),0\rangle, & \text{else if } \xi = 2, i \text{ even and } \exists k \leqslant i\,(\,e = e_1(k)\,); \\ \langle e,\xi\rangle, & \text{otherwise;} \end{cases}$$

$$\varphi_{b(i)}(\langle e,\xi\rangle) = \begin{cases} \langle e_1(i),\xi\rangle, & \text{if } e \in \{?, e_0\}; \\ \langle e,\xi\rangle, & \text{otherwise;} \end{cases}$$

$$W_{e_0} = \begin{cases} \mathrm{ran}(a[t_0]), & \text{if } t_0 \text{ is minimal with } \forall t \geqslant t_0\, N(a[t]) = N(a[t_0]); \\ \mathrm{ran}(a), & \text{no such } t_0 \text{ exists;} \end{cases}$$

$$W_{e_1(i)} = \mathrm{ran}(a[i]) \cup \{b(i)\} \cup \begin{cases} \{a(j)\} & \text{for first } j \geqslant i \text{ found} \\ & \text{with } \mathrm{MC}(i, a(j)); \\ \varnothing, & \text{no such } j \text{ exists;} \end{cases}$$

$$W_{e_2(i)} = \mathrm{ran}(a) \cup \{b(i)\}.$$

As the learner constantly puts out e_0 on every text for W_{e_0}, we have $W_{e_0} \in \mathcal{L}$. Thus, also N learns the finite language W_{e_0} and t_0 exists. Note that by the iterativeness of N we obtain $N(a[t_0]) = N(a[t_0]\frown a(i))$ for all $i \geqslant t_0$ and with this $N(a[t_0]\frown b(t_0)\frown \#^{\ell_{t_0}}) = N(a[t_0]\frown a(i)\frown b(t_0)\frown \#^{\ell_{t_0}})$ for all $i \geqslant t_0$.

$W_{e_1(t_0)}$ and $W_{e_2(t_0)}$ also lie in $\mathcal{L}$. To see that M explanatory learns both of them, note that, after having observed $b(t_0)$, M only changes its mind from $e_1(t_0)$ to $e_2(t_0)$ after having seen $a(i)$ and $a(j)$ with $i, j \geqslant t_0$ and $i \in 2\mathbb{N}$ as well as $j \in 2\mathbb{N}+1$. This clearly happens for every text for the infinite language $W_{e_2(t_0)}$. As $|W_{e_1(t_0)} \setminus (\operatorname{content}(a[t_0]) \cup \{b(t_0)\})| \leqslant 1$, this mind change never occurs for any text for $W_{e_1(t_0)}$.

The syntactic non-U-shapedness of M's learning processes can be easily seen as for all $k, l \in \mathbb{N}$ the languages W_{e_0}, $W_{e_1(k)}$ and $W_{e_2(l)}$ are pairwise distinct, the learner never returns to an abandoned hypothesis and M only leaves hypothesis $\langle e_1(k), 0\rangle$ for $\langle e_1(k), \xi\rangle$, $\xi \neq 0$, if $W_{e_1(k)}$ is not correct.

Next, we show the existence of $j \geqslant t_0$ with $\operatorname{MC}(t_0, a(j))$. Assume towards a contradiction that j does not exist. Then $W_{e_1(t_0)} = \operatorname{content}(a[t_0]) \cup \{b(t_0)\}$. As M learns this language from the text $a[t_0]\frown b(t_0)\frown \#^\infty$, so does N. The convergence of N implies the existence of ℓ_{t_0}. Thus, for every $j \in \mathbb{N}$ we either have $N(a[t_0]\frown b(t_0)\frown \#^{\ell_{t_0}}\frown a(j)) = N(a[t_0]\frown b(t_0)\frown \#^{\ell_{t_0}})$ or the computation of $N(a[t_0]\frown b(t_0)\frown \#^{\ell_{t_0}}\frown a(j))$ does not terminate. Because N is iterative and learns $W_{e_2(t_0)}$, it may not be undefined and therefore always the latter is the case. But then N will not learn $W_{e_1(t_0)}$ and $W_{e_2(t_0)}$ as they are different but N does not make a mind-change on the text $a[t_0]\frown b(t_0)\frown \#^{\ell_{t_0}}\frown a$ after having observed the initial segment $a[t_0]\frown b(t_0)\frown \#^{\ell_{t_0}}$, due to its iterativeness. Hence, j exists and $W_{e_1(t_0)} = \operatorname{ran}(a[t_0]) \cup \{b(t_0), a(j)\}$.

Finally, by the choice of j, the learner N does perform a syntactic U-shape on the text $a[t_0]\frown a(j)\frown b(t_0)\frown \#^{\ell_{t_0}}\frown a(j)\frown \#^\infty$ for $W_{e_1(t_0)}$. More precisely, t_0 and ℓ_{t_0} were chosen such that $N(a[t_0]\frown a(j)\frown b(t_0)\frown \#^{\ell_{t_0}})$ has to be correct and the characterizing property of j assures

$$N(a[t_0]\frown a(j)\frown b(t_0)\frown \#^{\ell_{t_0}}) \neq N(a[t_0]\frown a(j)\frown b(t_0)\frown \#^{\ell_{t_0}}\frown a(j)).$$

Thus, no iterative learner can explanatory syntactically non-U-shapedly learn the language $\mathcal{L}$. □

By Corollary 44 we also obtain $[\mathbf{ItTxtSDecEx}] \subsetneq [\mathbf{TxtBMS_*SDecEx}]$ and $[\mathbf{ItTxtConvEx}] \subsetneq [\mathbf{TxtBMS_*ConvEx}]$.

5 Related Open Problems

We have given a complete map for learning with bounded memory states, where, on the way to success, the learner must use only finitely many states. Future work can address the complete maps for learning with an a priori bounded number of memory states, which needs very different combinatorial arguments. Results in this regard can be found in [CCJS07] and [CK13]. We expect to see trade-offs, for example allowing for more states may make it possible to add various learning

restrictions (just as non-deterministic finite automata can be made deterministic at the cost of an exponential state explosion).

Also memory-restricted learning from positive and negative data (so-called informant) has only partially been investigated for iterative learners and not at all for other models of memory-restricted learning. Very interesting also in regard of 1-1 hypothesis spaces that prevent coding tricks is the **Bem**-hierarchy, see [FJO94], [LZ96] and [CJLZ99].

Acknowledgements

This work was supported by DFG Grant Number KO 4635/1-1. We are grateful to the people supporting us.

References

AKS18. M. Aschenbach, T. Kötzing, and K. Seidel. Learning from informants: Relations between learning success criteria. *arXiv preprint arXiv:1801.10502*, 2018.

Ang80. D. Angluin. Inductive inference of formal languages from positive data. *Information and control*, 45(2):117–135, 1980.

BB75. L. Blum and M. Blum. Toward a mathematical theory of inductive inference. *Information and Control*, 28:125–155, 1975.

BCM+08. G. Baliga, J. Case, W. Merkle, F. Stephan, and R. Wiehagen. When unlearning helps. *Information and Computation*, 206:694–709, 2008.

Cas74. J. Case. Periodicity in generations of automata. *Mathematical Systems Theory*, 8(1):15–32, 1974.

Cas94. J. Case. Infinitary self-reference in learning theory. *Journal of Experimental and Theoretical Artificial Intelligence*, 6:3–16, 1994.

CC13. L. Carlucci and J. Case. On the necessity of U-shaped learning. *Topics in Cognitive Science*, 5:56–88, 2013.

CCJS07. L. Carlucci, J Case, S. Jain, and F. Stephan. Results on memory-limited U-shaped learning. *Information and Computation*, 205:1551–1573, 2007.

CJLZ99. J. Case, S. Jain, S. Lange, and T. Zeugmann. Incremental concept learning for bounded data mining. *Information and Computation*, 152:74–110, 1999.

CK10. J. Case and T. Kötzing. Strongly non-U-shaped learning results by general techniques. In Adam Tauman Kalai and Mehryar Mohri, editors, *COLT 2010*, pages 181–193, 2010.

CK13. J. Case and T. Kötzing. Memory-limited non-u-shaped learning with solved open problems. *Theoretical Computer Science*, 473:100–123, 2013.

CK16. J. Case and T. Kötzing. Strongly non-u-shaped language learning results by general techniques. *Information and Computation*, 251:1–15, 2016.

CM08. J. Case and S. Moelius. U-shaped, iterative, and iterative-with-counter learning. *Machine Learning*, 72:63–88, 2008.

CM11. J. Case and S. Moelius. Optimal language learning from positive data. *Information and Computation*, 209:1293–1311, 2011.

FJO94. M. Fulk, S. Jain, and D. Osherson. Open problems in Systems That Learn. *Journal of Computer and System Sciences*, 49(3):589–604, December 1994.

Gol67. E. Gold. Language identification in the limit. *Information and Control*, 10:447–474, 1967.

Jan91. K. P. Jantke. Monotonic and nonmonotonic inductive inference of functions and patterns. In *Nonmonotonic and Inductive Logic, 1st International Workshop, Proc.*, pages 161–177, 1991.

JKMS16. S. Jain, T. Kötzing, J. Ma, and F. Stephan. On the role of update constraints and text-types in iterative learning. *Information and Computation*, 247:152–168, 2016.

JLZ07. S. Jain, S. Lange, and S. Zilles. Some natural conditions on incremental learning. *Information and Computation*, 205:1671–1684, 2007.

JMZ13. S. Jain, S. Moelius, and S. Zilles. Learning without coding. *Theoretical Computer Science*, 473:124–148, 2013.

JORS99. S. Jain, D. Osherson, J. Royer, and A. Sharma. *Systems that Learn: An Introduction to Learning Theory*. MIT Press, Cambridge, Massachusetts, second edition, 1999.

KP16. T. Kötzing and R. Palenta. A map of update constraints in inductive inference. *Theoretical Computer Science*, 650:4–24, 2016.

KS16. T. Kötzing and M. Schirneck. Towards an atlas of computational learning theory. In *33rd Symposium on Theoretical Aspects of Computer Science*, 2016.

KSS17. T. Kötzing, M. Schirneck, and K. Seidel. Normal forms in semantic language identification. In *Proc. of Algorithmic Learning Theory*, pages 493–516. PMLR, 2017.

LZ96. S. Lange and T. Zeugmann. Incremental learning from positive data. *Journal of Computer and System Sciences*, 53:88–103, 1996.

MPU+92. G. Marcus, S. Pinker, M. Ullman, M. Hollander, T.J. Rosen, and F. Xu. *Overregularization in Language Acquisition*. Monographs of the Society for Research in Child Development, vol. 57, no. 4. University of Chicago Press, 1992. Includes commentary by H. Clahsen.

Odi99. P. Odifreddi. *Classical Recursion Theory*, volume II. Elsivier, Amsterdam, 1999.

OSW82. D. Osherson, M. Stob, and S. Weinstein. Learning strategies. *Information and Control*, 53:32–51, 1982.

OSW86. D. Osherson, M. Stob, and S. Weinstein. *Systems that Learn: An Introduction to Learning Theory for Cognitive and Computer Scientists*. MIT Press, Cambridge, Mass., 1986.

RC94. J. Royer and J. Case. *Subrecursive Programming Systems: Complexity and Succinctness*. Research monograph in *Progress in Theoretical Computer Science*. Birkhäuser Boston, 1994.

SS82. S. Strauss and R. Stavy, editors. *U-Shaped Behavioral Growth*. Developmental Psychology Series. Academic Press, NY, 1982.

Wie91. R. Wiehagen. A thesis in inductive inference. In *Nonmonotonic and Inductive Logic, 1st International Workshop, Proc.*, pages 184–207, 1991.