Memory-Limited U-Shaped Learning

Lorenzo Carlucci[1,⋆], John Case[2,⋆⋆], Sanjay Jain[3,⋆⋆⋆], and Frank Stephan[4,†]

[1] Department of Computer and Information Sciences, University of Delaware, Newark, DE 19716-2586, USA and Dipartimento di Matematica, Università di Siena, Pian dei Mantellini 44, Siena, Italy, EU
carlucci5@unisi.it
[2] Department of Computer and Information Sciences, University of Delaware, Newark, DE 19716-2586, USA
case@cis.udel.edu
[3] School of Computing, 3 Science Drive 2, National University of Singapore, Singapore 117543, Republic of Singapore
sanjay@comp.nus.edu.sg
[4] School of Computing and Department of Mathematics, National University of Singapore, 3 Science Drive 2, Singapore 117543, Republic of Singapore
fstephan@comp.nus.edu.sg

Abstract. U-shaped learning is a learning behaviour in which the learner first *learns* something, then *unlearns* it and finally *relearns* it. Such a behaviour, observed by psychologists, for example, in the learning of past-tenses of English verbs, has been widely discussed among psychologists and cognitive scientists as a fundamental example of the non-monotonicity of learning. Previous theory literature has studied whether or not U-shaped learning, in the context of Gold's formal model of learning languages from positive data, is *necessary* for learning some tasks.

It is clear that human learning involves memory limitations. In the present paper we consider, then, this question of the necessity of U-shaped learning for some learning models featuring *memory limitations*. Our results show that the question of the necessity of U-shaped learning in this memory-limited setting depends on delicate tradeoffs between the learner's ability to remember its own previous conjecture, to store some values in its long-term memory, to make queries about whether or not items occur in previously seen data *and* on the learner's choice of hypothesis space.

1 Introduction and Motivation

U-Shaped learning. *U-shaped learning* occurs when the learner first learns a correct behaviour, then abandons that correct behaviour and finally returns to it once again. This pattern of learning has been observed by cognitive and developmental psychologists in a variety of child development phenomena, such as lan-

⋆ Supported in part by NSF grant number NSF CCR-0208616.
⋆⋆ Supported in part by NSF grant number NSF CCR-0208616.
⋆⋆⋆ Supported in part by NUS grant number R252–000–127–112.
† Supported in part by NUS grant number R252–000–212–112.

G. Lugosi and H.U. Simon (Eds.): COLT 2006, LNAI 4005, pp. 244–258, 2006.

guage learning [6, 19, 24], understanding of temperature [24, 25], understanding of weight conservation [5, 24], object permanence [5, 24] and face recognition [7].

The case of language acquisition is paradigmatic. In the case of the past tense of English verbs, it has been observed that children learn correct syntactic forms (call/called, go/went), then undergo a period of overregularization in which they attach regular verb endings such as 'ed' to the present tense forms even in the case of irregular verbs (break/breaked, speak/speaked) and finally reach a final phase in which they correctly handle both regular and irregular verbs. This example of U-shaped learning behaviour has figured so prominently in the so-called "Past Tense Debate" in cognitive science that competing models of human learning are often judged on their capacity for modeling the U-shaped learning phenomenon [19, 22, 26].

Recent interest in U-shaped learning is witnessed by the fact that the *Journal of Cognition and Development* dedicated its first issue in the year 2004 to this phenomenon.

While the prior cognitive science literature on U-shaped learning was typically concerned with modeling *how* humans achieve U-shaped behaviour, [2, 8] are motivated by the question of *why* humans exhibit this seemingly inefficient behaviour. Is it a mere harmless evolutionary inefficiency or is it *necessary* for full human learning power? A technically answerable version of this question is: are there some formal learning tasks for which U-shaped behaviour is logically necessary? The answer to this latter question requires that we first describe some formal criteria of successful learning.

A learning machine **M** reads an infinite sequence consisting of the elements of any language L in arbitrary order with possibly some pause symbols # in between elements. During this process the machine outputs a corresponding sequence $e_0\, e_1\, \ldots$ of hypotheses (grammars) which may generate the language L to be learned. Sometimes, especially when numerically coded, we also call these hypotheses *indices*. A fundamental criterion of successful learning of a language is called *explanatory learning* (**Ex**-*learning*) and was introduced by Gold in [13]. Explanatory learning requires that the learner's output conjectures stabilize in the limit to a *single* conjecture (grammar/program, description/explanation) that generates the input language.

For each such criterion, a *non U-shaped learner* is naturally modeled as a learner that never *semantically* returns to a previously abandoned correct conjecture on languages it learns according to that criterion. It is shown in [2] that every **Ex**-learnable class of languages is **Ex**-learnable by a non U-shaped learner, that is, for **Ex**-learnability, U-shaped learning is *not* necessary. In [2], it is also noted that, by contrast, for behaviourally correct learning, U-shaped learning *is* necessary for full learning power. In [8] it is shown that, for non-trivial vacillatory learning, U-shaped learning is again necessary for full learning power.

Memory-Limited Learning. It is clear that human learning involves memory limitations. In the present paper we consider the necessity of U-shaped learning in formal *memory-limited* versions of language learning. In the prior literature at least the following three types of memory-limited learning have been studied.

A most basic concept of memory-limited learning is *iterative learning* [18, 28], according to which the learner reacts to its current data item, can remember its own last conjecture but cannot store *any* of the strictly previously seen data items.

Iterative learning admits of learning non-trivial classes. For example, the class of finite sets is iteratively learnable as is a class of self-describing sets, for example, the class of languages with the least element coding a grammar for the language. Furthermore, for each $m \geq 1$, the class of unions of m of Angluin's [1] pattern languages is iteratively learnable [11].

The criterion of *n-feedback learning* is a variant of iterative learning where, in addition, the learner can make n simultaneous queries asking whether some datum has been seen in the past [11, 18]. Finally, a learner is called an *n-bounded example memory* learner [11, 18, 21] if, besides reacting to its currently seen data item and remembering its own last conjecture, it is allowed to store in "long term memory" at most n strictly previously seen data items.

For the present paper, our first intention was to study the impact of forbidding U-shaped learning in each of the above three models of memory-limited learning. So far we have had success for these problems only for some more restricted variants of the three models. Hence, we now describe these variants.

Our variants of iterative learning are motivated by two aspects of Gold's model.

The first aspect is the absolute freedom allowed regarding the *semantic* relations between successive conjectures, and between the conjectures and the input. Many forms of semantic constraints on the learner's sequence of hypotheses have been studied in the previous literature (for example, conservativity [1], consistency [1, 3], monotonicity [15, 29], set-drivenness [27]) and it is reasonable to explore their interplay with U-shaped learning in the memory-bounded setting of iterative learning.

Secondly, it is well-known that the choice of the hypothesis space from which the learner can pick its conjectures has an impact on the learning power [17, 18]. We accordingly also consider herein U-shaped iterative learning with restrictions on the hypothesis space.

For the case of feedback learning, we introduce and consider a model called *n-memoryless feedback learning* which restricts n-feedback learning so that the learner does *not* remember its last conjecture. These criteria form a hierarchy of more and more powerful learning criteria increasing in n and, for $n > 0$, are incomparable to iterative learning. The criterion of 0-memoryless feedback learning is properly contained in the criterion of iterative learning.

Finally, we introduce a more limited variant of bounded example memory, *c-bounded memory states learning* for which the learner does not remember its previous conjecture *but* can store any one out of c different *values* in its long term memory [12, 16]. For example, when $c = 2^k$, the memory is equivalent to k bits of memory. By Theorem 16, these criteria form a hierarchy of more and more powerful learning criteria increasing in c. Furthermore, the comparisons between bounded memory states learning, iterative learning and memoryless feedback learning are presented in Remark 17.

Non U-Shaped Learning. Our main objective is to investigate the relations of above discussed notions of memory limited learning with respect to non U-shapedness. In Section 3 we investigate this question first with respect to *iterative learning* and state the major open problem whether non U-shapedness is restrictive for iterative learning. In this regard, Theorem 5 shows that U-shaped learning is necessary for the full learning power of class-preserving iterative learning [18].

In Section 4 we study, in the context of iterative learning, the relation of the non U-shapedness constraint to other well studied constraints on the *semantic* behaviour of the learner's conjectures. We consider *class-consistent learning* [1, 3], according to which the learner's conjectures, on the languages it learns, must generate all the data on which they are based. *Monotonic learning* by a machine **M** [29] requires that, on any input language L that **M Ex**-learns, a new hypothesis cannot reject an element $x \in L$ that a previous hypothesis already included. Theorem 9 shows that class-consistent iterative learners can be turned into iterative non U-shaped *and* monotonic learners.

In Section 5, we consider the impact of forbidding U-shaped learning for n-memoryless feedback learning. Theorem 12 shows that U-shaped learning *is necessary* for the full learning power of n-memoryless feedback learners.

In Section 6, Theorem 18 shows that U-shaped behaviour does *not* enhance the learning power of 2-bounded memory states learners, that is, learners having 1 bit of memory.

Note. Our results herein on memory-limited models are presented only for **Ex**-*learning*. Furthermore, because of space limitations, many proofs and some results have been omitted. We refer the reader to [9] for details.

2 Notation and Preliminaries

For general background on Recursion Theory and any unexplained recursion theoretic notation, we refer the reader to [20]. The symbol $\mathbb{N}$ denotes the set of natural numbers, $\{0, 1, 2, 3, \ldots\}$. Cardinality of a set S is denoted by $\mathrm{card}(S)$. $\mathrm{card}(S) \leq *$ denotes that S is finite. We let $\langle \cdot, \cdot \rangle$ stand for Cantor's computable, bijective mapping $\langle x, y \rangle = \frac{1}{2}(x+y)(x+y+1)+x$ from $\mathbb{N} \times \mathbb{N}$ onto $\mathbb{N}$. Note that $\langle \cdot, \cdot \rangle$ is monotonically increasing in both of its arguments.

By φ we denote a fixed *acceptable numbering* (programming system) for the partial-recursive functions mapping $\mathbb{N}$ to $\mathbb{N}$. By φ_i we denote the partial-recursive function computed by the program with number i in the φ-system. By Φ we denote an arbitrary fixed Blum complexity measure [4] for the φ-system. By W_i we denote the domain of φ_i. That is, W_i is then the recursively enumerable (r.e.) subset of $\mathbb{N}$ accepted by the φ-program i. The symbol $\mathcal{L}$ ranges over classes of r.e. sets and L, H range over r.e. sets. By $\overline{L}$, we denote the complement of L, that is $\mathbb{N} - L$. By $W_{i,s}$ we denote the set $\{x \leq s : \Phi_i(x) \leq s\}$.

Quite frequently used in this paper is the existence of a one-one recursive function $\mathrm{pad}(e, X)$ with $W_{\mathrm{pad}(e,X)} = W_e$, where — according to the context — X

might be a number, a finite set or a finite sequence. In particular, pad is chosen such that e, X can be computed from $\text{pad}(e, X)$ by a recursive function.

We now present concepts from language learning theory [13, 14]. A *sequence* σ is a mapping from an initial segment of $\mathbb{N}$ into $(\mathbb{N} \cup \{\#\})$. The empty sequence is denoted by λ. The *content* of a sequence σ, denoted $\text{content}(\sigma)$, is the set of natural numbers in the range of σ. The *length* of σ, denoted by $|\sigma|$, is the number of elements in σ. So, $|\lambda| = 0$. For $n \leq |\sigma|$, the initial sequence of σ of length n is denoted by $\sigma[n]$. So, $\sigma[0]$ is λ.

Intuitively, the pause-symbol # represents a pause in the presentation of data. We let σ, τ and γ range over finite sequences. We denote the sequence formed by the concatenation of τ at the end of σ by $\sigma\tau$. $(\mathbb{N} \cup \{\#\})^*$ denotes the set of all finite sequences.

A *text* T for a language L is a mapping from $\mathbb{N}$ into $(\mathbb{N} \cup \{\#\})$ such that L is the set of natural numbers in the range of T. $T(i)$ represents the $(i+1)$-th element in the text. The *content* of a text T, denoted by $\text{content}(T)$, is the set of natural numbers in the range of T; that is, the language which T is a text for. $T[n]$ denotes the finite initial sequence of T with length n. We now define the basic paradigm of learning in the limit, explanatory learning.

Definition 1. A learner $\mathbf{M} : (\mathbb{N} \cup \{\#\})^* \to (\mathbb{N} \cup \{?\})$ is a (possibly partial) recursive function which assigns hypotheses to finite strings of data. $\mathbf{M}$ **Ex**-learns a class $\mathcal{L}$ (equivalently $\mathbf{M}$ is an **Ex**-learner for $\mathcal{L}$) iff, for every $L \in \mathcal{L}$ and every text T for L, $\mathbf{M}$ is defined on all initial segments of T, and there is an index n such that $\mathbf{M}(T[n]) \neq ?$, $W_{\mathbf{M}(T[n])} = L$ and $\mathbf{M}(T[m]) \in \{\mathbf{M}(T[n]), ?\}$ for all $m \geq n$. **Ex** denotes the collection of all classes of languages that can be **Ex**-learned from text.

For **Ex**-learnability one may assume without loss of generality that the learner is total. However, for some of the criteria below, such as class consistency and iterative learning, this cannot be assumed without loss of generality. The requirement for $\mathbf{M}$ to be defined on each initial segment of each text for a language in $\mathcal{L}$ is also assumed for learners with other criteria considered below.

Now we define non U-shaped learning. A non U-shaped learner never makes the sequence correct–incorrect–correct while learning a language that it actually learns. Thus, since such a learner has eventually to be correct, one can make the definition a bit simpler than the idea behind the notion suggests.

Definition 2. [2] (a) We say that $\mathbf{M}$ is *non U-shaped* on text T, if $\mathbf{M}$ never makes a mind change from a conjecture for $\text{content}(T)$ to a conjecture for a different set.

(b) We say that $\mathbf{M}$ is non U-shaped on L if $\mathbf{M}$ is non U-shaped on each text for L. We say that $\mathbf{M}$ is non U-shaped on $\mathcal{L}$ if $\mathbf{M}$ is non U-shaped on each $L \in \mathcal{L}$.

(c) Let $\mathbf{I}$ be a learning criterion. Then **NUI** denotes the collection of all classes $\mathcal{L}$ such that there exists a machine $\mathbf{M}$ that learns $\mathcal{L}$ according to $\mathbf{I}$ and is non U-shaped on $\mathcal{L}$.

3 Iterative Learning

The **Ex**-model makes the assumption that the learner has access to the full history of previous data. On the other hand it is reasonable to think that humans have more or less severe memory limitations. This observation motivates, among other criteria discussed in the present paper, the concept of *iterative learning*. An iterative learner features a severe memory limitation: it can remember its own previous conjecture but not its *past* data items. Moreover, each conjecture of an iterative learner is determined as an algorithmic function of the previous conjecture *and* of the current input data item.

Definition 3. [27] An iterative learner is a (possibly partial) function $\mathbf{M} : (\mathbb{N} \cup \{?\}) \times (\mathbb{N} \cup \{\#\}) \to (\mathbb{N} \cup \{?\})$ together with an initial hypothesis $e_0 \in \mathbb{N} \cup \{?\}$. $\mathbf{M}$ **It**-learns a class $\mathcal{L}$ iff, for every $L \in \mathcal{L}$ and every text T for L, the sequence $e_0, e_1, \ldots$ defined inductively by the rule $e_{n+1} = \mathbf{M}(e_n, T(n))$ satisfies: there exists an m such that e_m is an index for L and for all $n \geq m$, $e_n \in \{e_m, ?\}$. **It** denotes the collection of all iteratively learnable classes.

For iterative learners (without other constraints), one may assume without loss of generality that they never output ?.

It is well-known that $\mathbf{It} \subset \mathbf{Ex}$ [28]. On the other hand, iterative learning is not restrictive for behaviourally correct learning. Thus, all our notions regarding iterative learning will be modifications of the basic **Ex**-learning paradigm.

In [2] the main question regarding the necessity of U-shaped behaviour in the context of **Ex**-learning was answered in the negative. It was shown that $\mathbf{Ex} = \mathbf{NUEx}$, meaning that every **Ex**-learnable class can be learned by a non U-shaped **Ex**-learner. However, non U-shaped learning is restrictive for behaviourally correct learning and vacillatory learning [8]. Similarly, non U-shaped learning *may* become restrictive when we put memory limitations on **Ex**-learning. Our main motivation for the results presented in this section is the following problem, which remains open.

Problem 4. *Is* $\mathbf{It} = \mathbf{NUIt}$*?*

Many results in the present work were obtained in order to approximate an answer to this open problem.

We now briefly recall some basic relations of iterative learning with two criteria of learning that feature, like non U-shaped learning, a semantic constraint on the learner's sequence of hypotheses.

The first such notion is *set-driven learning* [27], where the hypotheses of a learner on inputs σ, τ are the same whenever $\mathrm{content}(\sigma) = \mathrm{content}(\tau)$. We denote by **SD** the collection of all classes learnable by a set-driven learner. It is shown in [16, Theorem 7.7] that $\mathbf{It} \subset \mathbf{SD}$.

A criterion that implies non U-shapedness is *conservative learning* [1]. A learner is conservative iff whenever it make a mind change from a hypothesis i to j then it has already seen some datum $x \notin W_i$. **Consv** denotes the collection of all classes having a conservative learner.

It is shown in [16] that **SD** $\subseteq$ **Consv**, thus, **It** $\subset$ **Consv**. By definition, every hypothesis abandoned by a conservative learner is incorrect and thus **Consv** $\subseteq$ **NUEx** follows. It is well known that the latter inclusion is proper. The easiest way to establish it is to use Angluin's proper inclusion **Consv** $\subset$ **Ex** [1] and the equality from **Ex** = **NUEx** [2].

Normally, in Gold-style language learning, a learner outputs as hypotheses just indices from a fixed acceptable enumeration of all r.e. languages, since all types of output (programs, grammars and so on) can be translated into these indices. There have also been investigations [1, 17, 18] where the hypothesis space is fixed in the sense that the learner has to choose its hypotheses either from this fixed space (exact learning) or from a space containing exactly the same languages (class-preserving learning).

We introduce a bit of terminology (from [1]) to explain the notion. An infinite sequence $L_0, L_1, L_2, \ldots$ of recursive languages is called *uniformly recursive* if the set $\{\langle i, x\rangle : x \in L_i\}$ is recursive. A class $\mathcal{L}$ of recursive languages is said to be an *indexed family* of recursive languages if $\mathcal{L} = \{L_i : i \in \mathbb{N}\}$ for some uniformly recursive sequence $L_0, L_1, L_2, \ldots$; the latter is called a *recursive indexing* of $\mathcal{L}$. As indexed families are quite well-behaved, Angluin found a nice characterization for when an indexed family is explanatorily learnable and they became a frequent topic for the study of more restrictive notions of learnability as, for example, in [12, 17, 18].

Let $\mathcal{L}$ be an indexed family of recursive sets. We say that a machine **M** explanatorily identifies $\mathcal{L}$ using a hypothesis space $L_0, L_1, L_2, \ldots$ iff **M**, for every $L \in \mathcal{L}$ and for every text for L, **M** converges to some j such that $L = L_j$. The hypothesis space $L_0, L_1, L_2, \ldots$ is class preserving for $\mathcal{L}$ iff it contains all and only the languages in $\mathcal{L}$. In what follows, for a learning criterion **I**, $\mathbf{I}^{\mathrm{cp}}$ stands for class-preserving **I**-learning, the collection of all classes of languages that can be **I**-learned by some learner using a class-preserving hypothesis space.

Theorem 5. *There exists an indexed family in* $\mathbf{It}^{\mathrm{cp}} - \mathbf{NUEx}^{\mathrm{cp}}$.

The positive side can be done using an indexed (recursive) family as hypothesis space, whereas the diagonalization against negative side can be done for any r.e. class preserving hypothesis space.

4 Consistent and Monotonic Iterative Learning

Forbidding U-shapes is a *semantic* constraint on a learner's sequence of conjectures. In this section we study the interplay of this constraint with other well-studied semantic constraints, but in the memory-limited setting of iterative learning.

We now describe and then formally define the relevant variants of semantic constraints on the sequence of conjectures. *Consistent learning* was introduced in [3] (in the context of function learning) and essentially requires that the learner's conjectures do not contradict known data, *strong monotonic learning* was introduced in [15] and requires that semantically the learner's conjectures on

every text for any language (even the ones that the learner does *not* learn) are set-theoretically nondecreasing. *Monotonic learning*, as introduced in [29], relaxes the condition of strong-monotonicity by requiring that, for each language L that the learner actually learns, the intersection of L with the language generated by a learner's conjecture is a superset of the intersection of L with the language generated by any of the learner's previous conjectures.

Definition 6. [3, 15, 29] A learner $\mathbf{M}$ is *consistent* on a class $\mathcal{L}$ iff for all $L \in \mathcal{L}$ and all σ with content$(\sigma) \subseteq L$, $\mathbf{M}(\sigma)$ it defined and an index of a set containing content(σ). **Cons** denotes the collection of all classes which have a **Ex**-learner which is consistent on the class of all sets. **ClassCons** denotes the collection of all classes $\mathcal{L}$ which have a **Ex**-learner which is consistent on $\mathcal{L}$.

A learner $\mathbf{M}$ is strong monotonic iff $W_i \subseteq W_j$ whenever $\mathbf{M}$ outputs on any text for any language at some time i and later j. **SMon** denotes the collection of all classes having a strong monotonic **Ex**-learner.

A learner $\mathbf{M}$ for $\mathcal{L}$ is monotonic iff $L \cap W_i \subseteq L \cap W_j$ whenever $\mathbf{M}$ outputs on a text for some language $L \in \mathcal{L}$ at some time i and later j. **Mon** denotes the criterion of all classes having a monotonic **Ex**-learner.

Note that there are classes $\mathcal{L} \in$ **ClassCons** such that only partial learners witness this fact. Criteria can be combined. For example, **ItCons** is the criterion consisting of all classes which have an iterative and consistent learner. The indication of an oracle as in the criterion **ItConsSMon**$[K]$ below denotes that a learner for the given class must on the one hand be iterative, consistent and strong-monotonic while on the other hand the constraint of being recursive is weakened to the permission to access a halting-problem oracle for the inference process. The next result gives some basic connections between iterative, strongly monotonic and consistent learning.

Theorem 7. (a) **ItCons** $\subseteq$ **ItConsSMon**.
(b) **ConsSMon** $\subseteq$ **ItConsSMon**.
(c) **ItSMon** $\subseteq$ **NUIt**.
(d) **SMon** $\subseteq$ **ItConsSMon**$[K]$.

Proof. (a) Given an iterative consistent learner $\mathbf{M}$ for $\mathcal{L}$, let — as in the case of normal learners — $\mathbf{M}(\sigma)$ denote the hypothesis which $\mathbf{M}$ makes after having seen the sequence σ. Now define a recursive one-one function f such that, for every index e, $W_{f(e)} = \bigcup_{\sigma \in \{\sigma' : \mathbf{M}(\sigma') = e\}} \text{content}(\sigma)$. Since $\mathbf{M}$ is consistent, content$(\sigma) \subseteq W_{\mathbf{M}(\sigma)}$ for all σ and so $W_{f(e)} \subseteq W_e$. The new learner $\mathbf{N}$ is the modification of $\mathbf{M}$ which outputs $f(e)$ instead of e; $\mathbf{N}$ is consistent since whenever one can reach a hypothesis e through a string containing a datum x then $x \in W_{f(e)}$. Since f is one-one, $\mathbf{N}$ is also iterative and follows the update rule $\mathbf{N}(f(e), x) = f(\mathbf{M}(e, x))$.

It is easy to see that $\mathbf{N}$ is strongly monotonic: Assume that $\mathbf{M}(e, y) = e'$ and x is any element of $W_{f(e)}$. Then there is a σ with $\mathbf{M}(\sigma) = e$ and $x \in \text{content}(\sigma)$. It follows that $\mathbf{M}(\sigma y) = e'$, $x \in \text{content}(\sigma y)$ and $x \in W_{f(e')}$. So $W_{f(e)} \subseteq W_{f(e')}$ and the transitiveness of the inclusion gives the strong monotonicity of $\mathbf{N}$.

It remains to show that $\mathbf{N}$ learns $\mathcal{L}$. Let $L \in \mathcal{L}$ and T be a text for L and e be the index to which $\mathbf{M}$ converges on T. The learner $\mathbf{N}$ converges on T to $f(e)$. Since $W_e = L$ it holds that $W_{f(e)} \subseteq L$. Furthermore, for every n there is $m > n$ with $\mathbf{M}(T[m]) = e$, thus $T(n) \in W_{f(e)}$ and $L \subseteq W_{f(e)}$. This completes the proof of part (a).

(b) A consistent learner never outputs ?. Now, given a strong monotonic and consistent learner $\mathbf{M}$ for some class $\mathcal{L}$, one defines a recursive one-one function $f : (\mathbb{N} \cup \{\#\})^* \to \mathbb{N}$ such that

$$W_{f(\sigma)} = W_{\mathbf{M}(\sigma)} \cup \text{content}(\sigma)$$

and initializes a new iterative learner $\mathbf{N}$ with the hypothesis $f(\lambda)$ and the following update rule for the hypothesis $f(\sigma)$ and observed datum x:

- If $\mathbf{M}(\sigma x) = \mathbf{M}(\sigma)$ then $\mathbf{N}(f(\sigma), x) = f(\sigma)$;
- If $\mathbf{M}(\sigma x) \neq \mathbf{M}(\sigma)$ then one takes the length-lexicographic first extension τ of σx such that $W_{\mathbf{M}(\eta),|\sigma|} \subseteq \text{content}(\tau)$, for all $\eta \preceq \sigma$, and defines $\mathbf{N}(f(\sigma), x) = f(\tau)$.

Note that in the second case, $\text{content}(\tau) = \text{content}(\sigma x) \cup (\bigcup_{\eta \preceq \sigma} W_{\mathbf{M}(\eta),|\sigma|})$ and that the length-lexicographic ordering is just taken to single out the first string with this property with respect to some ordering. The new iterative learner is strongly monotonic since whenever it changes the hypothesis then it does so from $f(\sigma)$ to $f(\tau)$, for some τ extending σ, and thus $W_{f(\sigma)} = \text{content}(\sigma) \cup W_{\mathbf{M}(\sigma)} \subseteq \text{content}(\tau) \cup W_{\mathbf{M}(\tau)} = W_{f(\tau)}$ as $\mathbf{M}$ is strong monotonic. Furthermore, $\mathbf{N}$ is also consistent: whenever it sees a number x outside $W_{f(\sigma)}$ then x is also outside $W_{\mathbf{M}(\sigma)}$ and $\mathbf{M}(\sigma x) \neq \mathbf{M}(\sigma)$ by the consistency of $\mathbf{M}$. Then the new τ constructed contains x explicitly and therefore $x \in W_{\mathbf{N}(f(\sigma),x)}$. By the strong monotonicity of $\mathbf{N}$, an element once incorporated into a hypothesis is also contained in all future hypotheses. So it remains to show that $\mathbf{N}$ actually learns $\mathcal{L}$.

Given $L \in \mathcal{L}$ and a text T for L, there is a sequence of strings $\sigma_0, \sigma_1, \ldots$ such that $\sigma_0 = \lambda$ and $\mathbf{N}(f(\sigma_n), T(n)) = f(\sigma_{n+1})$. By induction one can show that $\sigma_n \in (L \cup \{\#\})^*$ and $W_{\mathbf{M}(\sigma_n)} \subseteq L$ for all n. There are two cases.

First, there is an n such that $\sigma_m = \sigma_n$ for all $m \geq n$. Then $L \subseteq W_{f(\sigma_n)}$ since $\mathbf{N}$ is a consistent learner and eventually converges to this hypothesis on the text L. Furthermore, $W_{f(\sigma_n)} \subseteq L$ as mentioned above, so $\mathbf{N}$ learns L.

Second, for every n there is an $m > n$ such that σ_m is a proper extension of σ_n. Let T' be the limit of all σ_n. One can easily see that T' contains data from two sources, some items taken over from T and some elements taken from sets $W_{\mathbf{M}(\eta)}$ with $\eta \preceq \sigma_n$ for some n; since $\mathbf{M}$ is strong monotonic these elements are all contained in L and so $\text{content}(T') \subseteq L$. Furthermore, for every n the element $T(n)$ is contained in $W_{f(\sigma_{n+1})}$ and thus there is an extension σ_k of σ_{n+1} which is so long that $T(n) \in W_{\mathbf{M}(\sigma_{n+1}),|\sigma_k|} \cup \text{content}(\sigma_{n+1})$. If then for some $m \geq k$ the string σ_{m+1} is a proper extension of σ_m, then $T(n) \in \text{content}(\sigma_{m+1})$. As a consequence, T' is a text for L on which $\mathbf{M}$ converges to a hypothesis e. Then, one has that for all sufficiently large m, where σ_{m+1} is a proper extension of σ_m, σ_{m+1} is actually an extension of $\sigma_m T(m)$ and $\mathbf{M}(\sigma_m T(m)) = \mathbf{M}(\sigma_m)$,

which would by construction enforce that **N** does not update its hypothesis and $\sigma_{m+1} = \sigma_m$. By this contradiction, the second case does not hold and the first applies, thus **M** learns $\mathcal{L}$. This completes the proof of part (b).

(c) follows from the definition and (d) can be proved using techniques similar to part (b). □

Thus, **ItCons** and **ConsSMon** are contained in **NUIt**. Regarding part (d) above, it can be shown that one can replace K only by oracles $A \geq_T K$. Thus K is the optimal oracle in part (d).

Note that the proof of Theorem 7 (a) needs that the learner is an **ItCons**-learner and not just an **ItClassCons**-learner. In the latter case, the inference process cannot be enforced to be strong-monotonic as the following example shows.

Example 8. *The class $\mathcal{L}$ containing the set $\{0, 2, 4, 6, 8, \ldots\}$ of even numbers and all sets $\{0, 2, 4, \ldots, 2n\} \cup \{2n+1\}$ with $n \in \mathbb{N}$ is in* **ItClassCons** $-$ **SMon**.

So class-consistent, iterative learners cannot be made strong monotonic, even with an oracle. However, the next result shows that they can still be made monotonic, *and*, simultaneously, non U-shaped.

Theorem 9. **ItClassCons** $\subseteq$ **NUItMon**.

5 Memoryless Feedback Learning

An iterative learner has a severe memory limitation: it can store no previously seen data. On the other hand, crucially, an iterative learner remembers its previous conjecture. In this section we introduce a model of learning in which the learner does *not* remember its last conjecture *and* can store no previous input data. The learner is instead allowed to make, at each stage of its learning process, n feedback queries asking whether some n data items have been previously seen. We call such learners *n-memoryless feedback learners*. Theorem 12 shows that U-shaped behaviour is necessary for the full learning power of n-memoryless feedback learning.

Definition 10. Suppose $n \geq 0$. An *n-memoryless feedback learner* **M** has as input one datum from a text. It then can make n-queries which are calculated from its input datum. These queries are as to whether some n data items were already seen previously in the text. From its input and the answers to these queries, it either outputs a conjecture or the ? symbol. That is, given a language L and a text T for L, $\mathbf{M}(T(k))$ is determined as follows: First, n-values $q_i(T(k)), i = 1, \ldots, n$, are computed. Second, n bits $b_i, i = 1, \ldots, n$ are determined and passed on to **M**, where each b_i is 1 if $q_i(T(k)) \in \text{content}(T[k])$ and 0 otherwise. Third, an hypothesis e_k is computed from $T(k)$ and the b_i's. **M** $\mathbf{MLF}_n$-learns L if, for all T for L, for **M** on T, there is an k such that $W_{e_k} = L$ and $e_m \in \{?, e_k\}$ for all $m > k$. $\mathbf{MLF}_n$ denotes the class of all classes learnable by a n-memoryless feedback learner.

Theorem 11. *For all $n > 0$,* $\mathbf{NUMLF}_{n+1} \not\subseteq \mathbf{MLF}_n$.

It can be shown that **It** and $\mathbf{MLF}_n$ are incomparable for all $n > 0$. The next result shows that non U-shaped n-memoryless feedback learners are strictly less powerful than unrestricted n-memoryless feedback learners.

Theorem 12. *For $n > 0$,* $\mathbf{NUMLF}_n \subset \mathbf{MLF}_n$.

Proof Sketch. Let $F(e) = \max(\{1 + \varphi_i(e) : i \leq e \text{ and } \varphi_i(e)\downarrow\} \cup \{0\})$. Note that F grows faster than any partial or total recursive function. Based on this function F one now defines the family $\mathcal{L} = \{L_0, L_1, L_2, \ldots\} \cup \{H_0, H_1, H_2, \ldots\}$ where

$$L_e = \{\langle e, x\rangle : x < F(e) \text{ or } x \text{ is even}\};$$
$$H_e = \{\langle e, x\rangle : x < F(e) \text{ or } x \text{ is odd}\}.$$

We first show that $\mathcal{L} \in \mathbf{MLF}_1$. Note that the learning algorithm cannot store the last guess due to its memory limitation but might output a '?' in order to repeat that hypothesis. The parameter e is visible from each current input except '#'. The algorithm is the following:

If the new input is # or if the input is $\langle e, x\rangle$ and the Feedback says that $\langle e, x+1\rangle$ has already appeared in the input earlier, then output ?. Otherwise, if input is $\langle e, x\rangle$ and $\langle e, x+1\rangle$ has not yet appeared in input, then output a canonical grammar for L_e (H_e) if x is even (odd).

Consider any text T for L_e. Let n be such that $\text{content}(T[n]) \supseteq L_e \cap \{\langle e, x\rangle : x \leq F(e) + 1\}$. Then, it is easy to verify that, the learner will either output ? or a conjecture for L_e beyond $T[n]$. On the other hand, for any even $x > F(e)$, if $T(m) = \langle e, x\rangle$, then the learner outputs a conjecture for L_e after having seen $T[m+1]$ (this happens infinitely often, by definition of L_e). Thus, the learner $\mathbf{MLF}_1$-identifies L_e. Similar argument applies for H_e. A detailed case analysis shows that $\mathcal{L} \notin \mathbf{NUMLF}_1$, see [9]. □

Proposition 13. $\mathbf{NUIt} \not\subseteq \mathbf{NUMLF_1}$.

Finally, an iterative *total* learner that can store one selected previous datum is called a $\mathbf{Bem}_1$-learner (1-bounded example memory learner) in [11, 21]. One can also consider a "memoryless" version of this concept, where a learner does not memorize its previous hypothesis, but, instead, memorizes one selected previous datum.

Proposition 14. $\mathbf{NUBem}_1 \not\subseteq \mathbf{NUMLF}_1$.

6 Bounded Memory States Learning

Memoryless feedback learners store no information about the past. Bounded memory states learners, introduced in this section, have no memory of previous conjectures but can store a bounded number of values in their long term memory.

This model allows one to separate the issue of a learner's ability to remember its previous conjecture from the issue of a learner's ability to store information about the previously seen input. Similar models of machines with bounded long term memory are studied in [16]. We now proceed with the formal definition.

Definition 15. [16] For $c > 0$, a c-bounded memory states learner is a (possibly partial) function

$$\mathbf{M} : \{0, 1, \ldots, c-1\} \times (\mathbb{N} \cup \#) \to (\mathbb{N} \cup \{?\}) \times \{0, 1, \ldots, c-1\}$$

which maps the old long term memory content plus a datum to the current hypothesis plus the new long term memory content. The long term memory has the initial value 0. There is no initial hypothesis.

$\mathbf{M}$ learns a class $\mathcal{L}$ iff, for every $L \in \mathcal{L}$ and every text T for L, there is a sequence $a_0, a_1, \ldots$ of long term memory contents and $e_0, e_1, \ldots$ of hypotheses and a number n such that, for all m, $a_0 = 0$, $W_{e_n} = L$, $\mathbf{M}(a_m, T(m)) = (e_m, a_{m+1})$ and $m \geq n \Rightarrow e_m \in \{?, e_n\}$. We denote by $\mathbf{BMS}_c$ the collection of classes learnable by a c-bounded memory states learner.

Theorem 16. *For all $c > 1$, $\mathbf{BMS}_{c-1} \subset \mathbf{BMS}_c$.*

Remark 17. One can generalize $\mathbf{BMS}_c$ to **ClassBMS** and **BMS**. The learners for these criteria use natural numbers as long term memory. For **ClassBMS** we have the additional constraint that for every text of a language inside the *learnt* class, there is a constant c depending on the text such that the value of the long term memory is never a number larger than c. For **BMS** the corresponding constraint applies to all texts for all sets, even those outside the class.

One can show that **ClassBMS** = **It**. Furthermore, a class is in **BMS** iff it has a confident iterative learner, that is, an iterative learner which converges on every text, whether this text is for a language in the class to be learned or not.

It is easy to see that $\bigcup_c \mathbf{BMS}_c \subset \mathbf{BMS} \subset \mathbf{ClassBMS}$. Furthermore, $\mathbf{MLF}_0 = \mathbf{BMS}_1 = \mathbf{NUMLF}_0 = \mathbf{NUBMS}_1$, which are nontrivial. One can also show that $\mathbf{MLF}_m$ and $\mathbf{BMS}_n$ are incomparable for all $m > 0$ and $n > 1$.

We now give the main result of the present section, showing that every 2-bounded memory states learner *can* be simulated by a non U-shaped one.

Theorem 18. $\mathbf{BMS}_2 \subseteq \mathbf{NUBMS}_2$.

Proof Sketch. Suppose $\mathbf{M}$ witnesses $\mathcal{L} \in \mathbf{BMS}_2$. We assume without loss of generality that $\mathbf{M}$ does not change its memory on input #, as otherwise we could easily modify $\mathbf{M}$ to work without any memory.

In the following, "$*$" stands for the case that the value does not matter and in all (legal) cases the same is done.

Define a function P such that $P(?) =?$ and, for $e \in \mathbb{N}$, $P(e)$ is an index of the set $W_{P(e)} = \bigcup_{s \in S(e)} W_{e,s}$ where $S(e)$ is the set of all s satisfying either (a) or ((b) and (c) and (d)) below:

(a) There exists an $x \in W_{e,s}$, $\mathbf{M}(1, x) = (*, 0)$;
(b) For all $x \in W_{e,s}$, $[\mathbf{M}(0, x) = (*, 1) \Rightarrow \mathbf{M}(1, x) = (?, 1)]$;

(c) There exists an $x \in W_{e,s}$, $\mathbf{M}(0,x) = (?,1)$ or for all $x \in W_{e,s}$, $\mathbf{M}(0,x) = (*,0)$;
(d) For all $x \in W_{e,s} \cup \{\#\}$, $[\mathbf{M}(0,x) = (j,*) \Rightarrow W_{e,s} \subseteq W_j \ \wedge \ W_{j,s} \subseteq W_e]$.

Now we define for all $m \in \{0,1\}$, $j \in \mathbb{N} \cup \{?\}$ and $x \in \mathbb{N} \cup \{\#\}$,

$$\mathbf{N}(m,x) = \begin{cases} (P(j),0), & \text{if } m = 0 \text{ and } \mathbf{M}(0,x) = (j,0); \\ (j,1), & \text{if } m = 0 \text{ and } ((\mathbf{M}(0,x) = (j,1) \text{ and } \mathbf{M}(1,x) = (?,*)) \\ & \text{or } (\mathbf{M}(0,x) = (*,1) \text{ and } \mathbf{M}(1,x) = (j,*) \text{ and } j \neq ?)); \\ (j,1), & \text{if } m = 1 \text{ and } \mathbf{M}(1,x) = (j,*). \end{cases}$$

A detailed case analysis shows that $\mathbf{N}$ $\mathbf{NUBMS}_2$-identifies $\mathcal{L}$, see [9]. □

7 Conclusions and Open Problems

Numerous results related to non U-shaped learning for machines with severe memory limitations were obtained. In particular, it was shown that

- there are class-preservingly iteratively learnable classes that cannot be learned without U-shapes by any iterative class-preserving learner (Theorem 5),
- class-consistent iterative learners for a class can be turned into iterative non U-shaped *and* monotonic learners for that class (Theorem 9),
- for all $n > 0$, there are n-memoryless feedback learnable classes that cannot be learned without U-shapes by any n-memoryless feedback learner (Theorem 12) and, by contrast,
- every class learnable by a 2-bounded memory states learner can be learned by a 2-bounded memory states learner without U-shapes (Theorem 18).

The above results are, in our opinion, interesting in that they show how the impact of forbidding U-shaped learning in the context of severely memory-limited models of learning is far from trivial. In particular, the tradeoffs that our results reveal between remembering one's previous conjecture, having a long-term memory, and being able to make feedback queries are delicate and perhaps surprising. The following are some of the main open problems.

- Is $\mathbf{NUIt} \subset \mathbf{It}$?
- Is $\mathbf{MLF}_1 \subseteq \mathbf{NUMLF}_n$, for $n > 1$?
- Is $\mathbf{BMS}_c \subseteq \mathbf{NUBMS}_c$, for $c > 2$?

Also, the question of the necessity of U-shaped behaviour with respect to the stronger memory-limited variants of **Ex**-learning (bounded example memory and feedback learning) from the previous literature [11, 18] remains wide open. Humans can remember *much* more than one bit and likely retain something of their prior hypotheses; furthermore, they have some access to knowledge of whether they've seen something before. Hence, the open problems of this section may prove interesting for cognitive science.

References

1. Dana Angluin. Inductive inference of formal languages from positive data. *Information and Control*, 45:117–135, 1980.
2. Ganesh Baliga, John Case, Wolfgang Merkle, Frank Stephan and Rolf Wiehagen. *When unlearning helps.* Manuscript, 2005. Preliminary version of the paper appeared at ICALP, Lecture Notes in Computer Science, 1853:844–855, 2000.
3. Janis Bārzdiņš. Inductive Inference of automata, functions and programs. *International Mathematical Congress, Vancouver*, pages 771–776, 1974.
4. Manuel Blum. A machine independent theory of the complexity of the recursive functions. *Journal of the Association for Computing Machinery* 14:322–336, 1967.
5. T. G. R. Bower. Concepts of development. In *Proceedings of the 21st International Congress of Psychology.* Presses Universitaires de France, pages 79–97, 1978.
6. Melissa Bowerman. Starting to talk worse: Clues to language acquisition from children's late speech errors. In S. Strauss and R. Stavy, editors, *U-Shaped Behavioral Growth.* Academic Press, New York, 1982.
7. Susan Carey. Face perception: Anomalies of development. In S. Strauss and R. Stavy, editors, *U-Shaped Behavioral Growth*, Developmental Psychology Series. Academic Press, pages 169–190, 1982.
8. Lorenzo Carlucci, John Case, Sanjay Jain and Frank Stephan. Non U-Shaped Vacillatory and Team Learning. *Algorithmic Learning Theory* 16th International Conference, ALT 2005, Singapore, October 2005, Proceedings. Lecture Notes in Artificial Intelligence, 3734:241–255, 2005.
9. Lorenzo Carlucci, John Case, Sanjay Jain and Frank Stephan. Memory-limited U-shaped learning (Long version of the present paper). TR51/05, School of Computing, National University of Singapore, 2005.
10. Lorenzo Carlucci, Sanjay Jain, Efim Kinber and Frank Stephan. Variations on U-shaped learning. *Eighteenth Annual Conference on Learning Theory*, Colt 2005, Bertinoro, Italy, June 2005, Proceedings. Lecture Notes in Arificial Intelligence, 3559:382–397, 2005.
11. John Case, Sanjay Jain, Steffen Lange and Thomas Zeugmann. Incremental Concept Learning for Bounded Data Mining. *Information and Computation*, 152(1):74–110, 1999.
12. Rusins Freivalds, Efim B. Kinber and Carl H. Smith. On the impact of forgetting on learning machines. *Journal of the ACM*, 42:1146–1168, 1995.
13. E. Mark Gold. Language identification in the limit. *Information and Control*, 10:447–474, 1967.
14. Sanjay Jain, Daniel Osherson, James Royer and Arun Sharma. *Systems that Learn: An Introduction to Learning Theory.* MIT Press, Cambridge, Mass., second edition, 1999.
15. Klaus-Peter Jantke. Monotonic and non-monotonic Inductive Inference, *New Generation Computing*, 8:349–360, 1991.
16. Efim Kinber and Frank Stephan. Language learning from texts: mind changes, limited memory and monotonicity. *Information and Computation*, 123:224–241, 1995.
17. Steffen Lange and Thomas Zeugmann. The learnability of recursive languages in dependence on the space of hypotheses. *GOSLER-Report*, 20/93. Fachbereich Mathematik und Informatik, TH Leipzig, 1993.
18. Steffen Lange and Thomas Zeugmann. Incremental Learning from Positive Data. *Journal of Computer and System Sciences*, 53:88–103, 1996.

19. Gary Marcus, Steven Pinker, Michael Ullman, Michelle Hollander, T. John Rosen and Fei Xu. *Overregularization in Language Acquisition.* Monographs of the Society for Research in Child Development, volume 57, no. 4. University of Chicago Press, 1992. Includes commentary by Harold Clahsen.
20. Piergiorgio Odifreddi. *Classical Recursion Theory.* North Holland, Amsterdam, 1989.
21. Daniel Osherson, Michael Stob and Scott Weinstein. *Systems that Learn: An Introduction to Learning Theory for Cognitive and Computer Scientists.* MIT Press, 1986.
22. Kim Plunkett and Virginia Marchman. U-shaped learning and frequency effects in a multi-layered perceptron: implications for child language acquisition. *Cognition*, 38(1):43–102, 1991.
23. James Royer. *A Connotational Theory of Program Structure.* Lecture Notes in Computer Science, 273. Springer, 1987.
24. Sidney Strauss and Ruth Stavy, editors. *U-Shaped Behavioral Growth.* Developmental Psychology Series. Academic Press, 1982.
25. Sidney Strauss, Ruth Stavy and N. Orpaz. The child's development of the concept of temperature. Manuscript, Tel-Aviv University. 1977.
26. Niels A. Taatgen and John R. Anderson. Why do children learn to say broke? A model of learning the past tense without feedback. *Cognition*, 86(2):123–155, 2002.
27. Kenneth Wexler and Peter W. Culicover. *Formal Principles of Language Acquisition.* MIT Press, 1980.
28. Rolf Wiehagen. Limes-Erkennung rekursiver Funktionen durch spezielle Strategien. *Journal of Information Processing and Cybernetics*, 12:93–99, 1976.
29. Rolf Wiehagen. A thesis in Inductive Inference. *Nonmonotonic and Inductive Logic, First International Workshop*, Lecture Notes in Artificial Intelligence, 543:184–207, 1990.