Incremental Learning with Temporary Memory

Sanjay Jain [a,1], Steffen Lange [b],
Samuel E. Moelius III [c,*], and Sandra Zilles [d,2]

[a] *Department of Computer Science, National University of Singapore, Singapore 117417, Republic of Singapore*

[b] *Fachbereich Informatik, Hochschule Darmstadt, Haardtring 100, 64295 Darmstadt, Germany*

[c] *Department of Computer & Information Sciences, University of Delaware, Newark, Delaware 19716, USA*

[d] *Department of Computer Science, University of Regina, Regina, Saskatchewan, Canada S4S 0A2*

Abstract

In the inductive inference framework of learning in the limit, a variation of the bounded example memory (Bem) language learning model is considered. Intuitively, the new model constrains the learner's memory not only in *how much* data may be retained, but also in *how long* that data may be retained. More specifically, the model requires that, if a learner commits an example x to memory in some stage of the learning process and this example x is not presented to the learner again thereafter, then the learner will forget x, i.e., there is some subsequent stage for which x *no longer* appears in the learner's memory. This model is called *temporary example memory* (Tem) learning.

Many interesting results concerning the Tem-learning model are presented. For example, there exists a class of languages that can be identified by memorizing $k+1$ examples in the Tem sense, but that can*not* be identified by memorizing k examples in the Bem sense. On the other hand, there exists a class of languages that can be identified by memorizing *just* 1 *example* in the Bem sense, but that can*not* be identified by memorizing *any number of examples* in the Tem sense.

Results are also presented concerning the special case of learning classes of *infinite* languages.

Key words: Inductive inference, formal languages, incremental learning

The following is a common scenario in machine learning. A learner is repeatedly fed elements from an incoming stream of data. From this data, the learner must eventually generate a hypothesis that correctly identifies the contents of this stream of data. This is the case, for example, in many applications of neural networks (see [Mit97]).

In many cases, it would be impractical for a learning algorithm to *reconsider* all previously seen data when forming a new hypothesis. Thus, such learners are often designed to work in an *incremental* fashion, considering only the most recently presented datum, and possibly a few previously seen data that the learner considers to be significant.

This scenario has been studied formally by Lange and Zeugmann [LZ96] in the context of Gold-style language learning [Gol67]. Their model is called *bounded example memory* (*Bem*) learning. Intuitively, as the learner is fed elements from the incoming stream of data, the learner is allowed to commit up to k of these elements to memory, where k is *a priori* fixed. The learner may change which such elements are stored in its memory at any given time. However, any newly committed element *must* come from the incoming stream of data, *and*, the number of such elements can never exceed k. Among the results presented in [LZ96] is: for each k, there is a class of languages that can be identified by memorizing $k+1$ examples, but that can*not* be identified by memorizing only k examples (Theorem 7 below). Further results on the *Bem*-learning model are obtained in [CJLZ99,CCJS07]. [3]

The *Bem*-learning model allows that any given example may be stored in the learner's memory *indefinitely*. However, it has been observed in various areas of machine learning that the length of time for which data may be stored in a learner's memory can have an effect upon the capabilities of that learner (e.g., in reinforcement learning [LM92,McC96,Bak02] and in neural networks [HS97]). This suggests in particular that being able to access all of the memorized data during the whole learning process may be necessary to achieve full learning potential. In this paper we study the extent to which this is true for *Bem*-learning.

* corresponding author
Email addresses: sanjay@comp.nus.edu.sg (Sanjay Jain), s.lange@fbi.h-da.de (Steffen Lange), moelius@cis.udel.edu (Samuel E. Moelius III), zilles@cs.uregina.ca (Sandra Zilles).
[1] Sanjay Jain was supported in part by NUS grant number R252-000-308-112.
[2] Sandra Zilles was supported by the Alberta Ingenuity Fund; this article was written while she worked with the Department of Computing Science at the University of Alberta.
[3] Osherson, Stob, and Weinstein [OSW86] appear to be the first to have considered memory-limited learning.

We consider a variation of the *Bem*-learning model in which the learner's memory is constrained not only in *how much* data may be stored, but also in *how long* that data may be stored if it is not refreshed in the learner's observations (i.e., if it is not repeated in the sequence of data shown to the learner). More specifically, we consider a model which requires that, if a learner commits an example x to memory in some stage of the learning process and x is not presented to the learner again thereafter, then there is some subsequent stage for which x *no longer* appears in the learner's memory. We call this new model *temporary example memory* (*Tem*) learning.

Many interesting results concerning the *Tem*-learning model are presented. For example, there exists a class of languages that can be identified by memorizing $k + 1$ examples in the *Tem* sense, but that can*not* be identified by memorizing k examples in the *Bem* sense (Theorem 15). Thus, being able to store $k + 1$ examples temporarily, can allow one to learn more than being able to store k example indefinitely. On the other hand, there exists a class of languages that can be identified by memorizing *just* 1 *example* in the *Bem* sense, but that can*not* be identified by memorizing *any number of examples* in the *Tem* sense (Theorem 17). Thus, being able to store just 1 example indefinitely, can allow one to learn more than being able to store any number of examples temporarily. Both classes of languages are indexable classes of languages. Hence, the differences between the *Tem*-learning model and the *Bem*-learning model are witnessed by classes of languages having a natural algorithmic structure.

Results are also presented concerning the special case of learning classes of *infinite* languages. In this case, however, a completely different picture emerges. In particular, any such class that can be identified by memorizing an arbitrary but finite number of examples in the *Bem* sense, can also be identified by memorizing an arbitrary but finite number of examples in the *Tem* sense (Theorem 23). Intuitively, this latter result says that, when learning classes of infinite languages, restriction to temporary memory is, in fact, *not* a proper restriction, provided that there is no bound on the size of the learner's memory.

In the context of learning classes of infinite languages, some problems remain open. These problems are stated formally in Section 4.

1 Preliminaries

Computability-theoretic concepts not covered below are treated in [Rog67].

$\mathbb{N}$ denotes the set of natural numbers, $\{0, 1, 2, \ldots\}$. $\mathbb{N}^+$ denotes the set of *positive* natural numbers, $\{1, 2, 3, \ldots\}$. Lowercase italicized letters (e.g., a,

b, c), with or without decorations, range over elements of $\mathbb{N}$, unless stated otherwise.

$\langle \cdot, \cdot \rangle$ denotes any pairing function such that $\langle \cdot, \cdot \rangle$ is non-decreasing in both its arguments.

A *language* is a subset of $\mathbb{N}$. Uppercase italicized letters (e.g., A, B, C), with or without decorations, range over languages. For each A, $\mathrm{Fin}(A)$ denotes the collection of all finite subsets of A. For each nonempty $A \subseteq \mathbb{N}$, $\min A$ denotes the minimum element of A. $\min \emptyset \stackrel{\text{def}}{=} \infty$. For each nonempty, finite $A \subseteq \mathbb{N}$, $\max A$ denotes the maximum element of A. $\max \emptyset \stackrel{\text{def}}{=} -1$. $\mathcal{L}$, with or without decorations, ranges over collections of languages.

Σ ranges over *alphabets*, i.e., nonempty, finite sets of symbols. Lowercase typewriter-font letters (e.g., $\mathtt{a}$, $\mathtt{b}$, $\mathtt{c}$) are used to denote alphabet symbols. For a symbol $\mathtt{a}$ and $n \in \mathbb{N}$, $\mathtt{a}^n$ denotes the string consisting of n repetitions of $\mathtt{a}$ (e.g., $\mathtt{a}^3 = \mathtt{aaa}$). For each string x, $|x|$ denotes the length of x, i.e., the number of symbols in x. For each Σ, Σ^* denotes the set of all strings whose symbols are drawn from Σ; Σ^+ denotes the set of all nonempty such strings.

In some cases, for ease of presentation, we treat languages as sets of strings over some alphabet Σ, rather than as sets of natural numbers. In such cases, we assume a computable bijection between Σ^* and $\mathbb{N}$.

Let $\#$ be a reserved symbol. For each language L, t *is a text for* $L \stackrel{\text{def}}{\Leftrightarrow} t = (x_i)_{i \in \mathbb{N}}$, where $\{x_i \mid i \in \mathbb{N}\} \subseteq \mathbb{N} \cup \{\#\}$, and $L = \{x_i \mid i \in \mathbb{N}\} - \{\#\}$. For each L, $Text(L)$ denotes the set of all texts for L. For each text $t = (x_i)_{i \in \mathbb{N}}$, $content(t) \stackrel{\text{def}}{=} \{x_i \mid i \in \mathbb{N}\} - \{\#\}$. For each text t, and each $n \in \mathbb{N}$, $t[n]$ denotes the initial segment of t of length n.

For each one-argument partial function ψ, and each $x \in \mathbb{N}$, $\psi(x)\downarrow$ denotes that $\psi(x)$ converges; $\psi(x)\uparrow$ denotes that $\psi(x)$ diverges. We use $\uparrow$ to denote the value of a divergent computation.

σ, with or without decorations, ranges over finite initial segments of texts for arbitrary languages. For each σ, $|\sigma|$ denotes the length of σ (equivalently, the size of the domain of σ). For each $\sigma = (x_i)_{i<n}$, $content(\sigma) \stackrel{\text{def}}{=} \{x_i \mid i < n\} - \{\#\}$. λ denotes the empty initial segment (equivalently, the everywhere divergent function). For each σ_0 and σ_1, $\sigma_0 \cdot \sigma_1$ denotes the concatenation of σ_0 and σ_1.

$(\varphi_p)_{p \in \mathbb{N}}$ denotes any fixed, acceptable numbering of all one-argument partial computable functions from $\mathbb{N}$ to $\mathbb{N}$. Φ denotes a fixed Blum complexity measure for φ. For each $i, s, x \in \mathbb{N}$,

$$\varphi_i^s(x) \stackrel{\text{def}}{=} \begin{cases} \varphi_i(x), \text{ if } [x < s \ \wedge \ \Phi_i(x) < s]; \\ \uparrow, \qquad \text{otherwise.} \end{cases} \tag{1}$$

For each $i, s \in \mathbb{N}$, $W_i^s \stackrel{\text{def}}{=} \{x \mid \varphi_i^s(x)\downarrow\}$. For each $i \in \mathbb{N}$, $W_i \stackrel{\text{def}}{=} \bigcup_{s\in\mathbb{N}} W_i^s$. For each $s \in \mathbb{N}$, $W_\uparrow \stackrel{\text{def}}{=} W_\uparrow^s \stackrel{\text{def}}{=} \emptyset$.

An *inductive inference machine (IIM)* is a partial computable function whose inputs are initial segments of texts, and whose outputs are elements of $\mathbb{N}$ [OSW86]. $\mathbf{M}$, with or without decorations, ranges over IIMs.

Definitions 1, 2, and 4 below introduce formally three of the Gold-style learning criteria of relevance to this paper. Therein, *Lim*, *Sdr*, and *It* are mnemonic for *limiting*, *set-driven*, and *iterative*, respectively. The first of these, *Lim*-learning (Definition 1 below), is the most fundamental. Intuitively, an IIM $\mathbf{M}$ is fed successively longer finite initial segments of a text for a target language L. $\mathbf{M}$ successfully identifies the language (from the given text) iff $\mathbf{M}$ converges to a hypothesis that correctly identifies the language (i.e., to a j such that $W_j = L$).

Definition 1 (Gold [Gol67]) (a) Let $\mathbf{M}$ be an IIM, and let L be a language. $\mathbf{M}$ *LimTxt–identifies* L iff, for each text $t \in Text(L)$, there exists $n \in \mathbb{N}$ such that $W_{\mathbf{M}(t[n])} = L$ and $\mathbf{M}(t[i]) = \mathbf{M}(t[n])$ for each $i \geq n$.

(b) Let $\mathbf{M}$ be an IIM, and let $\mathcal{L}$ be a class of languages. $\mathbf{M}$ *LimTxt–identifies* $\mathcal{L}$ iff, for each $L \in \mathcal{L}$, $\mathbf{M}$ *LimTxt*–identifies L.

(c) $\mathit{LimTxt} = \{\mathcal{L} \mid (\exists\mathbf{M})[\mathbf{M}\ \mathit{LimTxt}\text{–identifies } \mathcal{L}]\}$.

The *Lim*-learning model allows that an IIM consider the entire initial segment of text presented to it when forming a new hypothesis. Thus, the IIM may consider: the *order* in which elements appear within that initial segment, and the *multiplicity* with which they appear. The set-driven (*Sdr*) learning model (Definition 2 below) restricts this. In particular, the *Sdr*-learning model requires that an IIM consider only the *contents* of any initial segment, and *not* the order or multiplicity of the elements therein.

Definition 2 (Wexler and Culicover [WC80]) (a) Let $\mathbf{M}$ be an IIM, let L be a language, and let $M : \mathrm{Fin}(\mathbb{N}) \to \mathbb{N}$ be a partial computable function. $\mathbf{M}$ *SdrTxt–identifies* L *via* M iff (i) and (ii) below.

(i) $\mathbf{M}$ *LimTxt*–identifies L.

(ii) For each text $t \in Text(L)$, and each $i \in \mathbb{N}$, $\mathbf{M}(t[i]) = M\big(content(t[i])\big)$.

(b) Let $\mathbf{M}$ be an IIM, and let $\mathcal{L}$ be a class of languages. $\mathbf{M}$ *SdrTxt–identifies* $\mathcal{L}$ iff there exists M such that, for each $L \in \mathcal{L}$, $\mathbf{M}$ *SdrTxt*–identifies L via M.

(c) $\mathit{SdrTxt} = \{\mathcal{L} \mid (\exists\mathbf{M})[\mathbf{M}\ \mathit{SdrTxt}\text{–identifies } \mathcal{L}]\}$.

Fulk [Ful90] proved that set-driven learning is a proper restriction of Gold's original model of learning in the limit.

Theorem 3 (Fulk [Ful90]) *$SdrTxt \subset LimTxt$.*

Both of the preceding learning models allow that an IIM consider an *unbounded* number of elements when forming a new hypothesis. This does not seem practicable, in general, and motivates a desire for *memory limited* models of learning. Iterative (It) learning (Definition 4 below) is such a memory limited model. The It-model requires that an IIM consider *only* its most recently conjectured hypothesis, and the most recently occurring element of an initial segment of text. Thus, the IIM can*not*, in general, consider previously conjectured hypotheses, *nor* previously occurring elements of an initial segment of text.

Definition 4 (Wiehagen [Wie76]) (a) Let $\mathbf{M}$ be an IIM, let L be a language, let $M : \mathbb{N} \times \mathbb{N} \to \mathbb{N}$ be a partial computable function, and let $j_0 \in \mathbb{N}$. $\mathbf{M}$ *ItTxt–identifies L via* (M, j_0) iff (i) and (ii) below.

- (i) $\mathbf{M}$ *LimTxt*–identifies L.
- (ii) For each text $t \in Text(L)$, (α) through (γ) below.
  - (α) For each $i \in \mathbb{N}$, $\mathbf{M}(t[i])\downarrow$.
  - (β) $\mathbf{M}(t[0]) = j_0$.
  - (γ) For each $i \in \mathbb{N}$, $\mathbf{M}(t[i+1]) = M\Big(\mathbf{M}(t[i]), t(i)\Big)$.

(b) Let $\mathbf{M}$ be an IIM, and let $\mathcal{L}$ be a class of languages. $\mathbf{M}$ *ItTxt–identifies* $\mathcal{L}$ iff there exists (M, j_0) such that, for each $L \in \mathcal{L}$, $\mathbf{M}$ *ItTxt*–identifies L via (M, j_0).

(c) $ItTxt = \{\mathcal{L} \mid (\exists \mathbf{M})[\mathbf{M}\ ItTxt\text{–identifies } \mathcal{L}]\}$.

Kinber and Stephan [KS95] showed that every class in *ItTxt* can be identified in the limit by a set-driven IIM, but that the converse is *not* true.

Theorem 5 (Kinber and Stephan [KS95]) *$ItTxt \subset SdrTxt$.*

Note that, in Definition 4(b), the behavior of $\mathbf{M}$ on any text t for a language in $\mathcal{L}$ is completely determined by j_0 and the behavior of M on j_0 and t. Thus, when referring to an iterative (or iterative-like) learner, we will, in some cases, refer only to (M, j_0) and avoid mention of $\mathbf{M}$ altogether. We do so similarly for set-driven learners (Definition 2). For iterative-like learning criteria that we define below (Definitions 6 and 10), we do so in terms of such (M, j_0) directly. In all such cases, it will be evident how to construct an appropriate IIM $\mathbf{M}$ from (M, j_0).

2 Bounded example memory (Bem) learning

The following is a natural relaxation of It-learning called *k-bounded example-memory* (Bem_k) learning (Lange and Zeugmann [LZ96]). Recall that the It-learning model allows that an IIM consider the most recently occurring element of an initial segment of text, but *not* previously occurring elements. By contrast, the Bem_k-learning model allows that the IIM consider up to k such previously occurring elements, where $k \in \mathbb{N}^+$ is *a priori* fixed.

Definition 6 (Lange and Zeugmann [LZ96]) Let $k \in \mathbb{N}^+$ be fixed.

(a) Let $M : \big(\mathbb{N}\times\text{Fin}(\mathbb{N})\big)\times\mathbb{N} \to \mathbb{N}\times\text{Fin}(\mathbb{N})$ be a partial computable function, let $j_0 \in \mathbb{N}$, and let L be a language. (M, j_0) *Bem_kTxt–identifies* L iff, for each text $t = (x_i)_{i\in\mathbb{N}} \in Text(L)$, (i) through (iii) below.
  - (i) For each $i \in \mathbb{N}$, $M_i(t)\downarrow$, where $M_0(t) = \langle j_0, \emptyset\rangle$ and $M_{i+1}(t) = M\Big(M_i(t), x_i\Big) = \langle j_{i+1}, X_{i+1}\rangle$.
  - (ii) There exists $n \in \mathbb{N}$ such that $W_{j_n} = L$ and $j_i = j_n$ for each $i \geq n$.
  - (iii) For each $i \in \mathbb{N}$, $X_{i+1} \subseteq X_i \cup \{x_i\}$ and $|X_{i+1}| \leq k$, where $X_0 = \emptyset$.

(b) Let (M, j_0) be as in (a), and let $\mathcal{L}$ be a class of languages. (M, j_0) *Bem_kTxt–identifies* $\mathcal{L}$ iff, for each $L \in \mathcal{L}$, (M, j_0) Bem_kTxt–identifies L.

(c) $Bem_kTxt = \{\mathcal{L} \mid (\exists M, j_0)[(M, j_0)\ Bem_kTxt\text{–identifies } \mathcal{L}]\}$.

For the remainder, let $\pi_1^2(\langle j, X\rangle) = j$ and $\pi_2^2(\langle j, X\rangle) = X$, for each $j \in \mathbb{N}$ and $X \in \text{Fin}(\mathbb{N})$.

Note that Definition 6 allows an IIM to change the contents of its example memory infinitely often, even *after* it has converged to its final hypothesis. Thus, changing the contents of the example memory does *not* constitute a mind-change.

The classes $(Bem_kTxt)_{k\in\mathbb{N}^+}$ defined in Definition 6(d) above form a proper hierarchy, as stated in the following theorem.

Theorem 7 (Lange and Zeugmann [LZ96]) *For each* $k \in \mathbb{N}^+$, $Bem_kTxt \subset Bem_{k+1}Txt$.

A natural variation of Lange and Zeugmann's model is to eliminate the restriction on the number of examples that can be memorized, i.e., to allow that the IIM store an arbitrary number of examples in its memory. We call the resulting learning model Bem_*-learning.

$$\begin{array}{ccccccc} Bem_1Txt & \subset & Bem_2Txt & \subset & Bem_3Txt & \subset \cdots & Bem_*Txt \\ \cup & & \cup & & \cup & & \cup \\ Tem_1Txt & \subset & Tem_2Txt & \subset & Tem_3Txt & \subset \cdots & Tem_*Txt \end{array}$$

$$Bem_1Txt \not\subseteq Tem_*Txt$$

Fig. 1. Summary of the results of Section 3.

The formal definition of Bem_*-learning is obtained from Definition 6 by replacing Bem_k by Bem_* and by dropping the condition $|X_{i+1}| \leq k$ in (a)(iii).[4] This definition immediately implies the following.

Proposition 8 For each $k \in \mathbb{N}^+$, $Bem_kTxt \subseteq Bem_*Txt$.

Kinber and Stephan [KS95] studied a flexible notion of memory limited learning that subsumes our definition of Bem_*-learning. As an immediate consequence of their results, one obtains a characterization of Bem_*-learning in terms of set-driven learning (Definition 2 above). Recall that, with set-driven learning, the IIM can consider *neither* the order of the elements in the text, *nor* the multiplicity with which they appeared. However, the full set of previously seen examples is always accessible. The similarity to the definition of Bem_*-learning is obvious; nonetheless, the proof of the characterization is not completely straightforward. The reader is referred to [KS95] for details.

Theorem 9 (Kinber and Stephan [KS95]) $SdrTxt = Bem_*Txt \subset LimTxt$.

3 Temporary example memory (Tem) learning

This section introduces the temporary example memory (Tem) learning model. This model is a natural *restriction* of *Bem*-learning. It requires that, if a learner commits an example x to memory in some stage of the learning process, then there is some subsequent stage for which x *no longer* appears in the learner's memory, unless x occurs in the text presented to the learner infinitely often (informally speaking, unless the learner will always get another chance to refresh its memory about x).

[4] N.B. The Bem_*-learning model does *not* afford the same capabilities to a learner as those provided by the *Lim*-learning model. Since the examples are stored in the learner's memory as a *set*, the learner can*not* consider the *order* in which those elements appeared, *nor* the *multiplicity* with which they appeared.

Figure 1 summarizes the main results of this section, which include the following. Theorem 15 says that there exists a class of languages that can be identified by memorizing $k+1$ examples in the Tem sense, but that can*not* be identified by memorizing k examples in the Bem sense. On the other hand, Theorem 17 says that there exists a class of languages that can be identified by memorizing *just* 1 *example* in the Bem sense, but that can*not* be identified by memorizing *any number of examples* in the Tem sense.

The following is the formal definition of Tem_k-learning. Note the addition of part (a)(iv), as compared to Definition 6. [5]

Definition 10 Let $k \in \mathbb{N}^+$ be fixed.

(a) Let $M : \big(\mathbb{N}\times\mathrm{Fin}(\mathbb{N})\big)\times\mathbb{N} \to \mathbb{N}\times\mathrm{Fin}(\mathbb{N})$ be a partial computable function, let $j_0 \in \mathbb{N}$, and let L be a language. (M, j_0) *Tem_kTxt–identifies* L iff, for each text $t = (x_i)_{i\in\mathbb{N}} \in Text(L)$, (i) through (iv) below.
  (i) For each $i \in \mathbb{N}$, $M_i(t)\downarrow$, where $M_0(t) = \langle j_0, \emptyset\rangle$ and $M_{i+1}(t) = M\big(M_i(t), x_i\big) = \langle j_{i+1}, X_{i+1}\rangle$.
  (ii) There exists $n \in \mathbb{N}$ such that $W_{j_n} = L$ and $j_i = j_n$ for each $i \geq n$.
  (iii) For each $i \in \mathbb{N}$, $X_{i+1} \subseteq X_i \cup \{x_i\}$ and $|X_{i+1}| \leq k$, where $X_0 = \emptyset$.
  (iv) For each $i \in \mathbb{N}$, there exists $i' \geq i$ such that $x_i \notin X_{i'+1}$ or $x_i = x_{i'}$. [6]

(b) Let (M, j_0) be as in (a), and let $\mathcal{L}$ be a class of languages. (M, j_0) *Tem_kTxt–identifies* $\mathcal{L}$ iff, for each $L \in \mathcal{L}$, (M, j_0) Tem_kTxt–identifies L.

(c) $Tem_kTxt = \{\mathcal{L} \mid (\exists M, j_0)[(M, j_0)\ Tem_kTxt\text{–identifies } \mathcal{L}]\}$.

The preceding definition immediately implies the following.

Proposition 11 For each $k \in \mathbb{N}^+$, $Tem_kTxt \subseteq Bem_kTxt$.

The formal definition of Tem_*-learning is obtained from Definition 10 by replacing Tem_k by Tem_* and by dropping the condition $|X_{i+1}| \leq k$ in (a)(iii). Again, a few observations follow immediately.

Proposition 12 (a) For each $k \in \mathbb{N}^+$, $Tem_kTxt \subseteq Tem_*Txt$.
(b) $Tem_*Txt \subseteq Bem_*Txt$.

A seemingly more restrictive version of learning with temporary memory would require the learner to eventually forget every datum, independent of whether or not it is repeated in the text infinitely often. In other words, no

---

[5] For simplicity, Definition 10 allows that *when* an example is removed from memory be determined by the learner, as opposed to, say, by the environment. Technically, this gives the learner more control than absolutely necessary. However, this also makes the negative results obtained even more surprising (see, e.g., Theorem 17).

[6] Note that in the preliminary version [LMZ08] of this paper, condition (iv) in this definition was slightly different. However, the definitions are logically equivalent.

datum can stay in the learner's memory forever from any point in time on. Definition 13 below captures this idea. (Note the change in part (a)(iv) of Definition 13, as compared to Definition 10.) This model is less intuitive, however, as Proposition 14 below shows, it is in fact *equivalent* to that of Definition 10. The reason we introduce this version of the model is that it is slightly simpler to use in the proofs of most of our results. In fact, in the preliminary version [LMZ08] of this paper, this was the original definition of $\mathit{Tem}_k\mathit{Txt}$.

Definition 13 Let $k \in \mathbb{N}^+$ be fixed.

(a) Let $M : \big(\mathbb{N}\times\mathrm{Fin}(\mathbb{N})\big)\times\mathbb{N} \to \mathbb{N}\times\mathrm{Fin}(\mathbb{N})$ be a partial computable function, let $j_0 \in \mathbb{N}$, and let L be a language. (M, j_0) *$\mathit{FinTem}_k\mathit{Txt}$–identifies* L iff, for each text $t = (x_i)_{i\in\mathbb{N}} \in \mathit{Text}(L)$, (i) through (iv) below.
  - (i) For each $i \in \mathbb{N}$, $M_i(t)\downarrow$, where $M_0(t) = \langle j_0, \emptyset\rangle$ and $M_{i+1}(t) = M\big(M_i(t), x_i\big) = \langle j_{i+1}, X_{i+1}\rangle$.
  - (ii) There exists $n \in \mathbb{N}$ such that $W_{j_n} = L$ and $j_i = j_n$ for each $i \geq n$.
  - (iii) For each $i \in \mathbb{N}$, $X_{i+1} \subseteq X_i \cup \{x_i\}$ and $|X_{i+1}| \leq k$, where $X_0 = \emptyset$.
  - (iv) For each $i \in \mathbb{N}$, there exists $i' \geq i$ such that $x_i \notin X_{i'+1}$.

(b) Let (M, j_0) be as in (a), and let $\mathcal{L}$ be a class of languages. (M, j_0) *$\mathit{FinTem}_k\mathit{Txt}$–identifies* $\mathcal{L}$ iff, for each $L \in \mathcal{L}$, (M, j_0) $\mathit{FinTem}_k\mathit{Txt}$–identifies L.

(c) $\mathit{FinTem}_k\mathit{Txt} = \{\mathcal{L} \mid (\exists M, j_0)[(M, j_0)\ \mathit{FinTem}_k\mathit{Txt}\text{–identifies } \mathcal{L}]\}$.

The formal definition of FinTem_*-learning is obtained from Definition 13 by replacing FinTem_k by FinTem_* and by dropping the condition $|X_{i+1}| \leq k$ in (a)(iii).

The following proposition will be used implicitly in our subsequent proofs.

Proposition 14 (a) For each $k \in \mathbb{N}^+$, $\mathit{FinTem}_k\mathit{Txt} = \mathit{Tem}_k\mathit{Txt}$.
(b) $\mathit{FinTem}_*\mathit{Txt} = \mathit{Tem}_*\mathit{Txt}$.

Proof of Proposition. We give only the proof of part (a). Let $k \in \mathbb{N}^+$ be fixed. Clearly, $\mathit{FinTem}_k\mathit{Txt} \subseteq \mathit{Tem}_k\mathit{Txt}$. Thus, it suffices to show $\mathit{Tem}_k\mathit{Txt} \subseteq \mathit{FinTem}_k\mathit{Txt}$. Let $\mathcal{L} \in \mathit{Tem}_k\mathit{Txt}$ be fixed, and let (M, j_0) be such that (M, j_0) $\mathit{Tem}_k\mathit{Txt}$–identifies $\mathcal{L}$. Let M' be such that $M'_0(t) = j_0$ and, for each $j \in \mathbb{N}$, $X \in \mathrm{Fin}(\mathbb{N})$, and $x \in \mathbb{N} \cup \{\#\}$,

$$M'(\langle j, X\rangle, x) = \begin{cases} M(\langle j, X\rangle, x), & \text{if } x \notin X; \\ M(\langle j, X\rangle, \#), & \text{otherwise.} \end{cases} \tag{2}$$

It remains to show that (M', j_0) $\mathit{FinTem}_k\mathit{Txt}$–identifies $\mathcal{L}$. This follows from Claims 1 through 3 below.

Let $L \in \mathcal{L}$ and $t = (x_i)_{i\in\mathbb{N}} \in \mathit{Text}(L)$ be fixed. Let $\hat{t} = (\hat{x}_i)_{i\in\mathbb{N}}$ be such that, for each $i \in \mathbb{N}$,

$$\hat{x}_i = \begin{cases} x_i, \text{ if } x_i \notin \hat{X}_i, \text{ where } \hat{X}_i \text{ is obtained} \\ \qquad \text{by running } M \text{ on } \hat{x}_0, ..., \hat{x}_{i-1}; \\ \#, \text{ otherwise.} \end{cases} \tag{3}$$

Claim 1. $\hat{t} \in Text(L)$.

Proof of Claim. Clearly, $content(\hat{t}) \subseteq content(t) = L$. Thus, it suffices to show that $L \subseteq content(\hat{t})$. By way of contradiction, let $i \in \mathbb{N}$ be *least* such that $x_i \in L - content(\hat{t})$. Clearly, by (3), $x_i \in \hat{X}_i$. Since $\hat{X}_i \subseteq \{\hat{x}_0, ..., \hat{x}_{i-1}\}$, there must be an $i' < i$ such that $\hat{x}_{i'} = x_i$. Since $\hat{x}_{i'} = x_i \in L$ (and, thus, $\hat{x}_{i'} \neq \#$), by (3), $\hat{x}_{i'} = x_{i'}$. But then $i' < i$ and $x_{i'} = \hat{x}_{i'} = x_i \in L - content(\hat{t})$, contradicting the choice of i. □ (*Claim 1*)

Claim 2. For each $i \in \mathbb{N}$, $M'_i(t) = M_i(\hat{t})$.

Proof of Claim. Clearly, $M'_0(t) = \langle j_0, \emptyset \rangle = M_0(\hat{t})$. So, suppose, inductively, that $M'_i(t) = M_i(\hat{t})$. Consider the following two cases.

CASE $x_i \notin \hat{X}_i$. Then,

$$\begin{aligned} M'_{i+1}(t) &= M'\Big(M'_i(t), x_i\Big) && \{\text{immediate}\} \\ &= M'\Big(M_i(\hat{t}), x_i\Big) && \{\text{by the induction hypothesis}\} \\ &= M\Big(M_i(\hat{t}), x_i\Big) && \{\text{by (2) and } x_i \notin \hat{X}_i, \text{ where } M_i(\hat{t}) = \langle \hat{\jmath}_i, \hat{X}_i \rangle\} \\ &= M\Big(M_i(\hat{t}), \hat{x}_i\Big) && \{\text{by (3) and } x_i \notin \hat{X}_i, \text{ where } M_i(\hat{t}) = \langle \hat{\jmath}_i, \hat{X}_i \rangle\} \\ &= M_{i+1}(\hat{t}) && \{\text{immediate}\}. \end{aligned}$$

CASE $x_i \in \hat{X}_i$. Then,

$$\begin{aligned} M'_{i+1}(t) &= M'\Big(M'_i(t), x_i\Big) && \{\text{immediate}\} \\ &= M'\Big(M_i(\hat{t}), x_i\Big) && \{\text{by the induction hypothesis}\} \\ &= M\Big(M_i(\hat{t}), \#\Big) && \{\text{by (2) and } x_i \in \hat{X}_i, \text{ where } M_i(\hat{t}) = \langle \hat{\jmath}_i, \hat{X}_i \rangle\} \\ &= M\Big(M_i(\hat{t}), \hat{x}_i\Big) && \{\text{by (3) and } x_i \in \hat{X}_i, \text{ where } M_i(\hat{t}) = \langle \hat{\jmath}_i, \hat{X}_i \rangle\} \\ &= M_{i+1}(\hat{t}) && \{\text{immediate}\}. \end{aligned}$$

$\square$ (*Claim 2*)

Let $(X'_i)_{i\in\mathbb{N}}$ be as in the definition of $FinTem_kTxt$ for (M', j_0) and t.

Claim 3. For each $i \in \mathbb{N}$, there exists $i' \geq i$ such that $x_i \notin X'_{i'+1}$.

Proof of Claim. By way of contradiction, let $i \in \mathbb{N}$ be such that, for each $i' \geq i$, $x_i \in X'_{i'+1}$. Then, by Claim 2, for each $i' \geq i$, $x_i \in \hat{X}_{i'+1}$. Clearly, by (3), for each $i' \geq i$, $\hat{x}_{i'+1} \neq x_i$. Thus, x_i occurs only finitely often in $\hat{t}$. But then, there must exist $i' \geq i$ such that $x_i \notin \hat{X}_{i'+1}$ (a contradiction). $\square$ (*Claim 3*)

$\square$ (*Proposition 14*)

The following main results of this section provide more insights into the relation between the Tem-learning model and Bem-learning model. Note that the observed differences are witnessed by indexable classes of languages. A class of languages $\mathcal{L}$ is indexable iff (by definition) there exists a computable function $d : \mathbb{N} \times \mathbb{N} \to \{0, 1\}$ such that $\mathcal{L} = \{L_i \mid i \in \mathbb{N}\}$ where, for each $i \in \mathbb{N}$, $L_i = \{x \in \mathbb{N} \mid d(i, x) = 1\}$ [Ang80,LZZ08]. Many interesting *and natural* classes of languages are indexable. For example, the classes of regular and context free languages [HMU01] are each indexable.

Intuitively, the first main result of this section says that being able to store $k + 1$ examples temporarily, can, in some cases, allow one to learn more than being able to store k examples indefinitely.

Theorem 15 For each $k \in \mathbb{N}^+$, $Tem_{k+1}Txt - Bem_kTxt \neq \emptyset$.

Proof. Let $k \in \mathbb{N}^+$. For separating Tem_{k+1} and Bem_k we use a class that was already used in [LZ96] for the separation of Bem_{k+1} and Bem_k. We set $\Sigma = \{\mathtt{a}, \mathtt{b}\}$. For every $j, \ell_0, \ldots, \ell_k \in \mathbb{N}$, let

$$L_{(j,\ell_0,\ldots,\ell_k)} = \{\mathtt{a}^{j+1}\} \cup \{\mathtt{b}^z \mid z \leq j\} \cup \{\mathtt{b}^{\ell_0}, \ldots, \mathtt{b}^{\ell_k}\}. \quad (4)$$

By $\mathcal{L}_k$ we denote the class containing $L = \{\mathtt{b}\}^*$ and all the languages $L_{(j,\ell_0,\ldots,\ell_k)}$ for $j, \ell_0, \ldots, \ell_k \in \mathbb{N}$.

The following M witnesses $\mathcal{L}_k \in Tem_{k+1}Txt$. As long as no string in $\{\mathtt{a}\}^+$ occurs in the input text, M stores the $(k + 1)$ longest strings in $\{\mathtt{b}\}^*$ seen so far and outputs an index for L along with this set. If a string $x \in \{\mathtt{a}\}^+$ appears, M outputs an index for the minimal language $L' \in \mathcal{L}_k$ that contains x and the strings memorized in its example memory. Past that point, there is no need to store further examples, because the target language must be a superset of L'. Moreover, in case L' does not equal the target language, the missing strings in

$\{\mathtt{b}\}^*$ will appear in some subsequent stage. If such a missing string appears, M updates its current guess accordingly. We omit further details.

Next we prove that $\mathcal{L}_k \notin \mathit{Bem}_k\mathit{Txt}$. Suppose the converse, i.e., there is an IIM $\mathbf{M}'$ that $\mathit{Bem}_k\mathit{Txt}$-identifies $\mathcal{L}_k$. Since $\mathbf{M}'$ learns $L = \{\mathtt{b}\}^*$, there exists a finite sequence σ with $\mathit{content}(\sigma) \subseteq L$ such that, for each finite sequence σ' with $\mathit{content}(\sigma') \subseteq L$, $\pi_1^2\Big(\mathbf{M}'(\sigma \cdot \sigma')\Big) = \pi_1^2\Big(\mathbf{M}'(\sigma)\Big)$, i.e., σ is a locking sequence [BB75] for $\mathbf{M}'$ and L.

Let $d = \max\{|x| \mid x \in \mathit{content}(\sigma)\}$. Then simple combinatorial arguments verify the following claim.

Claim 1. There are $\ell_0, \ell_0', \ldots, \ell_k, \ell_k' \in \mathbb{N}^+$ such that (i) – (iii) are fulfilled:

(i) $\{\ell_0, \ldots, \ell_k\} \neq \{\ell_0', \ldots, \ell_k'\}$.
(ii) $|\{\ell_0, \ldots, \ell_k\}| = |\{\ell_0', \ldots, \ell_k'\}| = k+1$.
(iii) $\pi_2^2\Big(\mathbf{M}'\Big(\sigma \cdot (\mathtt{b}^{d+\ell_0}, \ldots, \mathtt{b}^{d+\ell_k})\Big)\Big) = \pi_2^2\Big(\mathbf{M}'\Big(\sigma \cdot (\mathtt{b}^{d+\ell_0'}, \ldots, \mathtt{b}^{d+\ell_k'})\Big)\Big)$.

Proof of Claim. Let $c = |\mathit{content}(\sigma)|$ and $n = (k+1)^{k+2}(c+1)^k + 1$.

Firstly, consider the collection $\mathcal{D}$ of all sets $D = \mathit{content}(\sigma) \cup \{\mathtt{b}^{d+z_0}, \mathtt{b}^{d+z_1}, \ldots, \mathtt{b}^{d+z_k}\}$, where $1 \leq z_0 < z_1 < \cdots < z_k \leq n$. Obviously, $|\mathcal{D}| = \binom{n}{k+1}$.

Secondly, consider the collection $\mathcal{S}$ of all sets S of cardinality at most k with $S \subseteq \mathit{content}(\sigma) \cup \{\mathtt{b}^{d+z_0}, \mathtt{b}^{d+z_1}, \ldots, \mathtt{b}^{d+z_k}\}$, where again $1 \leq z_0 < z_1 < z_2 < \cdots < z_k \leq n$. Obviously, $|\mathcal{S}| = \sum_{j=0}^{k} \binom{c+n}{j}$.

Note that

$$
\begin{aligned}
& (k+1)^{k+2}(c+1)^k + 1 > (k+1)^{k+2}(c+1)^k \\
\Rightarrow\quad & n > (k+1)^{k+2}(c+1)^k \\
\Rightarrow\quad & (\tfrac{n}{k+1})^{k+1} > (k+1)(c+1)^k n^k \\
\Rightarrow\quad & (\tfrac{n}{k+1})^{k+1} > (k+1)(c+n)^k \\
\Rightarrow\quad & \binom{n}{k+1} > \textstyle\sum_{j=0}^{k} \binom{c+n}{j} \\
\Rightarrow\quad & |\mathcal{D}| > |\mathcal{S}|.
\end{aligned}
$$

Furthermore, σ is a locking sequence for $\mathbf{M}'$ and L, and $\mathbf{M}'$ can store at most k strings in its example memory. Therefore, there exist indices $\ell_0, \ell_0', \ldots, \ell_k, \ell_k' \in \mathbb{N}^+$ such that (i) – (iii) are fulfilled. This proves the claim. □ (*Claim 1*)

Finally we show that $\mathbf{M}'$ cannot identify all languages in $\mathcal{L}_k$. Let $\ell_0, \ell_0', \ldots, \ell_k, \ell_k' \in \mathbb{N}^+$ be fixed such that (i) – (iii) are fulfilled. We set $\hat{L}$ and $\tilde{L}$ as follows.

$$\hat{L} = \{\mathtt{a}^{d+1}\} \cup \{\mathtt{b}^z \mid z \leq d\} \cup \{\mathtt{b}^{d+\ell_0}, \ldots, \mathtt{b}^{d+\ell_k}\}. \tag{5}$$
$$\tilde{L} = \{\mathtt{a}^{d+1}\} \cup \{\mathtt{b}^z \mid z \leq d\} \cup \{\mathtt{b}^{d+\ell'_0}, \ldots, \mathtt{b}^{d+\ell'_k}\}. \tag{6}$$

Obviously $\hat{L}, \tilde{L} \in \mathcal{L}_k$ and $\hat{L} \neq \tilde{L}$. Let t be any text for $\{\mathtt{b}^z \mid z \leq d\}$, $\hat{t} = \sigma \cdot (\mathtt{b}^{d+\ell_0}, \ldots, \mathtt{b}^{d+\ell_k}) \cdot (\mathtt{a}^{d+1}) \cdot t$, and $\tilde{t} = \sigma \cdot (\mathtt{b}^{d+\ell'_0}, \ldots, \mathtt{b}^{d+\ell'_k}) \cdot (\mathtt{a}^{d+1}) \cdot t$. By construction, $\hat{t}$ is a text for $\hat{L}$ and $\tilde{t}$ is a text for $\tilde{L}$. Moreover, $\mathbf{M}'\Big(\sigma \cdot (\mathtt{b}^{d+\ell_0}, \ldots, \mathtt{b}^{d+\ell_k})\Big) = \mathbf{M}'\Big(\sigma \cdot (\mathtt{b}^{d+\ell'_0}, \ldots, \mathtt{b}^{d+\ell'_k})\Big)$. Consequently, if $\mathbf{M}'$ converges on both texts, the final conjecture returned by $\mathbf{M}'$ is the same for both texts. Thus, $\mathbf{M}'$ fails to learn at least one of the languages $\hat{L}$ and $\tilde{L}$. □ (*Theorem 15*)

Theorem 15 has the following consequences.

Corollary 16 (a) For each $k \in \mathbb{N}^+$, $\mathit{Tem}_k\mathit{Txt} \subset \mathit{Tem}_{k+1}\mathit{Txt}$.
(b) $\mathit{Tem}_*\mathit{Txt} - \bigcup_{k\in\mathbb{N}^+} \mathit{Bem}_k\mathit{Txt} \neq \emptyset$.
(c) $\bigcup_{k\in\mathbb{N}^+} \mathit{Bem}_k\mathit{Txt} \subset \mathit{Bem}_*\mathit{Txt}$.
(d) $\bigcup_{k\in\mathbb{N}^+} \mathit{Tem}_k\mathit{Txt} \subset \mathit{Tem}_*\mathit{Txt}$.

Proof of Corollary. (a) follows from Proposition 11 and Theorem 15. (b) is obtained using the tagged union of the classes used in the proof of Theorem 15, i.e.,

$$\Big\{\{\langle i, x\rangle \mid x \in L\} \mid L \in \mathcal{L}_i \ \wedge \ i \in \mathbb{N}\Big\}. \tag{7}$$

(c) follows from (b), and from Propositions 8 and 12(b). (d) follows from (b), and from Propositions 11 and 12(a). □ (*Corollary 16*)

In contrast to Theorem 15, restriction to temporary memory can have a significant effect upon a learner's capabilities, as demonstrated by our next main result. Intuitively, this result says that being able to store just 1 example indefinitely, can allow one to learn more than being able to store any number of examples temporarily.

Theorem 17 $\mathit{Bem}_1\mathit{Txt} - \mathit{Tem}_*\mathit{Txt} \neq \emptyset$.

Proof. Let $(D_i)_{i\in\mathbb{N}}$ be a canonical enumeration of $\mathrm{Fin}(\mathbb{N})$.

Let $\mathit{cyl}_i = \{\langle i, x\rangle \mid x \in \mathbb{N}\}$.

For each $i, k, r \in \mathbb{N}$, let $R_{i,k,r} = \{\langle i+1, x\rangle \mid x \in D_k\} \cup \{\langle 0, k\rangle, \langle i+1, r\rangle\}$.

Note that $\mathcal{L}_1 = \{R_{i,k,r} \mid i, k \in \mathbb{N} \ \wedge \ [r \in D_k \ \vee \ r > \max D_k]\}$ is an indexable class.

In addition, we will define nonempty sets $X_i \subseteq \mathit{cyl}_{i+1}$. We will define a decision procedure for X_i effectively in i. Thus, it will be easily seen that $\mathcal{L}_2 = \{X_i \mid i \in \mathbb{N}\}$ is an indexable class. The class $\mathcal{L} = \mathcal{L}_1 \cup \mathcal{L}_2$ will be the diagonalizing class.

The following learner will Bem_1Txt learn $\mathcal{L}$. The memory of the learner is the maximal element of the form $\langle i+1, r\rangle$ that is seen in the input. As long as the input is a subset of cyl_{i+1}, the learner will output X_i as its conjecture. If the learner ever sees an element of the form $\langle 0, k\rangle$ in the input, then from then on it will output the language $R_{i,k,r}$, where $\langle i+1, r\rangle$ is the memorized element at the time of the output.

We now define X_i. X_i initially contains $\langle i+1, 0\rangle$. Let σ_0 be the sequence containing just $\langle i+1, 0\rangle$. Let $x_0 = 1$. Intuitively, we determine whether $\langle i+1, x\rangle$ is a member of X_i, for all $x < x_s$, before going to stage s. σ_s will contain elements defined to be in X_i before stage s. Let $(\mathfrak{M}_i)_{i\in\mathbb{N}}$ be an algorithmic enumeration of all Tem_*-learners.

Go to stage 0.

Begin stage s.

0. Define more and more elements to be *not* in X_i until (if ever) an m is found such that $\mathfrak{M}_i(\sigma_s \cdot \#^m)\downarrow$ and

$$(\pi_2^2 \circ \mathfrak{M}_i)(\sigma_s \cdot \#^m) = \emptyset. \tag{8}$$

   Let $\sigma'_s = \sigma_s \cdot \#^m$, and $x'_s = x+1$, where x is maximal such that membership for $\langle i+1, x\rangle$ in X_i has been decided above. Let $\langle i+1, x\rangle$ be *not* in X_i for any x such that $x_s \leq x < x'_s$.
1. Let X_i contain $\langle i+1, x'_s\rangle$.
2. Define more and more elements to be *not* in X_i, until (if ever) an m' is found such that $\mathfrak{M}_i(\sigma'_s \cdot \langle i+1, x'_s\rangle \cdot \#^{m'})\downarrow$ and

$$(\pi_1^2 \circ \mathfrak{M}_i)(\sigma'_s) \neq (\pi_1^2 \circ \mathfrak{M}_i)(\sigma'_s \cdot \langle i+1, x'_s\rangle \cdot \#^{m'}). \tag{9}$$

   Let $\sigma_{s+1} = \sigma'_s \cdot \langle i+1, x'_s\rangle \cdot \#^{m'}$, and $x_{s+1} = x+1$, where x is maximal such that membership for $\langle i+1, x\rangle$ in X_i has been decided above. Let $\langle i+1, x\rangle$ be *not* in X_i for any x such that $x'_s < x < x_{s+1}$.
3. Proceed to stage $s+1$.

End stage s.

Note that if there are infinitely many stages then clearly $\mathfrak{M}_i$ does not identify X_i, as there are infinitely many conjecture changes on the text $\bigcup_{s\in\mathbb{N}} \sigma_s$ for X_i (via step 2). So, suppose that some stage s does not complete. If step 0 is not exited in stage s, then X_i is finite, and the memory of $\mathfrak{M}_i$ never becomes the empty set on the text $\sigma_s \cdot \#^\infty$ in step 0 above.

On the other hand, if step 2 is not exited, then consider $\mathfrak{M}_i$'s behavior on $\sigma'_s \cdot \langle i+1, x'_s \rangle \cdot \#^\infty$. As this is a text for X_i, $\mathfrak{M}_i$'s memory should eventually become the empty set. Note that the memory of $\mathfrak{M}_i$ on σ'_s also equals the empty set. Furthermore, $\mathfrak{M}_i$ does not change its conjecture beyond σ'_s on $\sigma'_s \cdot \langle i+1, x'_s \rangle \cdot \#^\infty$ (as step 2 did not succeed). Thus, $\mathfrak{M}_i$ does not remember whether it saw the input $\langle i+1, x'_s \rangle$ or not.

Let $k \in \mathbb{N}$ be such that $D_k = \{x \mid \langle i+1, x \rangle \in \mathit{content}(\sigma'_s)\}$. (Note that $x'_s > \max D_k$, and $\langle i+1, x'_s \rangle \notin \mathit{content}(\sigma'_s)$.) By the preceding observations, $\mathfrak{M}_i$ will converge to the same conjecture on texts $\sigma'_s \cdot \#^{m''} \cdot \langle 0, k \rangle \cdot \#^\infty$ and $\sigma'_s \cdot \langle i+1, x'_s \rangle \cdot \#^{m''} \cdot \langle 0, k \rangle \cdot \#^\infty$ (or diverge on both), for some m''. Thus, $\mathfrak{M}_i$ cannot distinguish between $R_{i,k,0}$ and R_{i,k,x'_s}. □ (*Theorem 17*)

Theorem 17 has the following consequences.

Corollary 18 (a) For each $k \in \mathbb{N}^+$, $\mathit{Tem}_k\mathit{Txt} \subset \mathit{Bem}_k\mathit{Txt}$.
(b) $\mathit{Tem}_*\mathit{Txt} \subset \mathit{Bem}_*\mathit{Txt}$.

Proof of Corollary. To show (a): by Proposition 11, $\mathit{Tem}_k\mathit{Txt} \subseteq \mathit{Bem}_k\mathit{Txt}$; by Theorems 7 and 17, and by Proposition 12(a), $\mathit{Bem}_k\mathit{Txt} - \mathit{Tem}_k\mathit{Txt} \neq \emptyset$. To show (b): by Proposition 12(b), $\mathit{Tem}_*\mathit{Txt} \subseteq \mathit{Bem}_*\mathit{Txt}$; by Proposition 8 and Theorem 17, $\mathit{Bem}_*\mathit{Txt} - \mathit{Tem}_*\mathit{Txt} \neq \emptyset$. □ (*Corollary 18*)

4 *Tem*-learning of classes of ***infinite*** languages

In this section, we consider the special case of *Tem*-learning of classes of infinite languages. Our main result of this section, Theorem 23, says that any class of infinite languages that can be identified by memorizing an arbitrary but finite number of examples in the *Bem* sense, can also be identified by memorizing an arbitrary but finite number of examples in the *Tem* sense.

Our first result of this section says that one of the important separation results obtained in Section 3 is witnessed by a class of infinite languages.

Theorem 19 For each $k \in \mathbb{N}^+$, there exists a class $\mathcal{L}_k$ of infinite languages such that $\mathcal{L}_k \in \mathit{Tem}_{k+1}\mathit{Txt} - \mathit{Bem}_k\mathit{Txt}$.

Proof. Let $k \in \mathbb{N}^+$. Fix $\Sigma = \{\mathtt{a}, \mathtt{b}, \mathtt{c}\}$. The witnessing class can be defined by taking the class $\mathcal{L}_k$ used in the proof of Theorem 15 and by adding the infinite set $\{\mathtt{c}\}^*$ to every language in this class. Further details are omitted.
□ (*Theorem 19*)

Before presenting our next main result, it is worth recalling the following.

Theorem 20 (Osherson, Stob, and Weinstein [OSW86]) Let $\mathcal{L}$ be any class of infinite languages. Then, $\mathcal{L} \in \mathit{LimTxt}$ iff $\mathcal{L} \in \mathit{SdrTxt}$.

Note that Theorems 9 and 20 have the following corollary.

Corollary 21 (of Theorems 9 and 20) Let $\mathcal{L}$ be any class of infinite languages. Then, $\mathcal{L} \in \mathit{LimTxt}$ iff $\mathcal{L} \in \mathit{Bem}_*\mathit{Txt}$.

Thus, Bem_*-learning is *not* a proper restriction when learning classes of infinite languages. This is in contrast to Theorem 9 which also says that Bem_*-learning *is* a proper restriction when learning classes of arbitrary languages.

Our next main result says that Tem_*-learning is equivalent to Bem_*-learning when learning classes of infinite languages. Thus, by Corollary 21, Tem_*-learning is similarly *not* a proper restriction when learning classes of infinite languages.

The proof of the aforementioned result requires the following technical lemma.

Lemma 22 Let L be a language. Suppose that M SdrTxt-identifies L and that $t \in \mathit{Text}(L)$. Then, there exists $i \in \mathbb{N}$ such that

$$\Big(\forall A \in \mathrm{Fin}(\mathbb{N})\Big)\Big[\mathit{content}(t[i]) \subseteq A \subseteq L \;\Rightarrow\; M(A)\!\downarrow = M\Big(\mathit{content}(t[i])\Big)\Big]. \tag{10}$$

Proof. It is straightforward to show that, if such an i did *not* exist, then one could construct another text t' for L on which M would never reach a final conjecture. □ (*Lemma 22*)

Theorem 23 Let $\mathcal{L}$ be any class of infinite languages. Then, $\mathcal{L} \in \mathit{Bem}_*\mathit{Txt}$ iff $\mathcal{L} \in \mathit{Tem}_*\mathit{Txt}$.

Proof. By Proposition 12(b), it suffices to show that, for each class of infinite languages $\mathcal{L}$, if $\mathcal{L} \in \mathit{Bem}_*\mathit{Txt}$, then $\mathcal{L} \in \mathit{Tem}_*\mathit{Txt}$. So, let $\mathcal{L}$ be a class of infinite languages, and suppose that $\mathcal{L} \in \mathit{Bem}_*\mathit{Txt}$. By Theorem 9, there exists M such that M SdrTxt-identifies $\mathcal{L}$. An M' is constructed such that M' $\mathit{Tem}_*\mathit{Txt}$-identifies $\mathcal{L}$. Intuitively, M' simulates M in such a way that: each element committed to memory by M' is removed after it has appeared in the language conjecture by M.

For ease of presentation, we argue that M' $\mathit{Tem}_*\mathit{Txt}$-identifies $\mathcal{L}$ on texts that do *not* contain #. For texts that *do* contain #, one can imagine that M' simply ignores each such occurrence.

$A_0 = B_0 = X_0 = \emptyset$ and $k_0 = 0$. For each $i \in \mathbb{N}$, $A_{i+1} = A_i$, $B_{i+1} = B_i$, $k_{i+1} = k_i$, and $X_{i+1} = X_i$, unless stated otherwise.

let $B_i^* = \begin{cases} B_i, & \text{if } k_i = 0; \\ W_{M(A_i)}^{s_i^{\min}}, & \text{if } k_i = 1, \text{ where } s_i^{\min} = \min\{s \mid (W_{M(A_i)}^{s+1} \cap X_i) \neq \emptyset\}; \end{cases}$

`/*` For the latter case, the proof of Claim 2 shows that $s_i^{\min} < \infty$. `*/`

let $X_i^+ = (X_i \cup \{x_i\})$;

let $C_i = (B_i^* \cup X_i^+)$;

let $S_i = \Big\{ s \le \max(C_i) \mid \quad B_i^* \subseteq W_{M(A_i)}^s \subseteq C_i \wedge \Big(W_{M(A_i)}^{s+1} \cap (X_i^+ - W_{M(A_i)}^s)\Big) \neq \emptyset \Big\}$;

if $(\exists A')\Big[B_i \subseteq A' \subseteq C_i \ \wedge\ \varphi_{p_M}^{\max(C_i)}(A'){\downarrow} \neq \varphi_{p_M}^{\max(C_i)}(A_i){\downarrow}\Big]$ **then**

    $A_{i+1} \leftarrow$ any such A'; $B_{i+1} \leftarrow C_i$; $k_{i+1} \leftarrow 0$; $X_{i+1} \leftarrow \emptyset$;

else if $S_i \neq \emptyset$ **then**

    $k_{i+1} \leftarrow 1$; $X_{i+1} \leftarrow (X_i^+ - W_{M(A_i)}^{s_i^{\max}})$, where $s_i^{\max} = \max(S_i)$;

else

    $X_{i+1} \leftarrow X_i^+$;

end if.

Fig. 2. The behavior of M' in the proof of Theorem 23.

Without loss of generality, suppose that $M(\emptyset){\downarrow}$. Let $p_M \in \mathbb{N}$ be such that, for each finite $A \subseteq \mathbb{N}$, $\varphi_{p_M}(A) = M(A)$. By 1-1 s-m-n [Rog67], there exists a 1-1 computable function f such that, for each finite $A, B \subseteq \mathbb{N}$, and each $k \in \{0, 1\}$, $W_{f(A,B,k)} = W_{M(A)}$.

For each $L \in \mathcal{L}$, each $t = (x_i)_{i \in \mathbb{N}} \in \mathit{Text}(L)$, and each $i \in \mathbb{N}$, let M' be as follows. $M'_0(t) = \langle f(\emptyset, \emptyset, 0), \emptyset \rangle$ and $M'_{i+1}(t) = M'\Big(\langle f(A_i, B_i, k_i), X_i \rangle, x_i\Big) = \langle f(A_{i+1}, B_{i+1}, k_{i+1}), X_{i+1} \rangle$, where A_{i+1}, B_{i+1}, k_{i+1}, and X_{i+1} are determined as in Figure 2.

Let $L \in \mathcal{L}$ and $t = (x_i)_{i \in \mathbb{N}} \in \mathit{Text}(L)$ be such that $(\forall i)[x_i \neq \#]$. For each $i \in \mathbb{N}$, let B_i^*, X_i^+, C_i, etc. be as in Figure 2.

That M' $\mathit{Tem}_*\mathit{Txt}$-identifies L from t follows from Claims 7 and 9 below, and from the definition of f.

Claim 1. For each $i \in \mathbb{N}$, if $(B_i^* \cup X_i) = \mathit{content}(t[i])$, then $C_i = \mathit{content}(t[i+1])$.

Proof of Claim. Immediate by the definition of C_i. □ (*Claim 1*)

Claim 2. For each $i \in \mathbb{N}$, (a) and (b) below.

(a) $(B_i^* \cup X_i) = \mathit{content}(t[i])$.
(b) $k_i = 1 \;\Rightarrow\; [i > 0 \;\wedge\; s_i^{\min} = s_{i-1}^{\max} \in S_i]$.

Proof of Claim. The proof is by induction on i. For the case when $i = 0$, $k_i = 0$ and $B_i \cup X_i = \emptyset = \mathit{content}(t[0])$. So, suppose that (a) and (b) hold for i. To show that (a) and (b) hold for $i+1$, consider the following cases.

Case (I) $(\exists A')\big[B_i \subseteq A' \subseteq C_i \;\wedge\; \varphi_{p_M}^{\max(C_i)}(A'){\downarrow} \neq \varphi_{p_M}^{\max(C_i)}(A_i){\downarrow}\big]$. Then, $A_{i+1} = A'$, $B_{i+1} = C_i$, $k_{i+1} = 0$, and $X_{i+1} = \emptyset$ (line 5 of Figure 2). Thus,

$$\begin{aligned}(B_{i+1}^* \cup X_{i+1}) &= (B_{i+1} \cup X_{i+1}) && \{\text{because } k_{i+1} = 0\}\\ &= (C_i \cup X_{i+1}) && \{\text{because } B_{i+1} = C_i\}\\ &= (C_i \cup \emptyset) && \{\text{because } X_{i+1} = \emptyset\}\\ &= C_i && \{\text{immediate}\}\\ &= \mathit{content}(t[i+1]) && \{\text{by (a) for } i \text{ and Claim 1}\}.\end{aligned} \tag{11}$$

Case (II) $[\neg(\text{I}) \;\wedge\; S_i \neq \emptyset]$. Then, $A_{i+1} = A_i$, $B_{i+1} = B_i$, $k_{i+1} = 1$, and $X_{i+1} = (X_i^+ - W_{M(A_i)}^{s_i^{\max}})$, where $s_i^{\max}$ is *largest* such that

$$s_i^{\max} \leq \max(C_i) \wedge B_i^* \subseteq W_{M(A_i)}^{s_i^{\max}} \subseteq C_i \wedge \left(W_{M(A_i)}^{s_i^{\max}+1} \cap (X_i^+ - W_{M(A_i)}^{s_i^{\max}})\right) \neq \emptyset \tag{12}$$

(line 7 of Figure 2).

To show that $s_{i+1}^{\min} \leq s_i^{\max}$ $(< \infty)$:

$$\begin{aligned}&(W_{M(A_{i+1})}^{s_i^{\max}+1} \cap X_{i+1})\\ &= (W_{M(A_i)}^{s_i^{\max}+1} \cap X_{i+1}) && \{\text{because } A_{i+1} = A_i\}\\ &= \left(W_{M(A_i)}^{s_i^{\max}+1} \cap (X_i^+ - W_{M(A_i)}^{s_i^{\max}})\right) && \{\text{because } X_{i+1} = (X_i^+ - W_{M(A_i)}^{s_i^{\max}})\}\\ &\neq \emptyset && \{\text{by (12)}\}.\end{aligned} \tag{13}$$

To show that $s_i^{\max} \leq s_{i+1}^{\min}$, note that, if $s_{i+1}^{\min} < s_i^{\max}$, then $s_{i+1}^{\min} + 1 \leq s_i^{\max}$ and, thus,

$$W^{s^{\min}_{i+1}+1}_{M(A_i)} \subseteq W^{s^{\max}_i}_{M(A_i)} \;\wedge\; (W^{s^{\min}_{i+1}+1}_{M(A_i)} \cap X_{i+1}) \neq \emptyset \;\wedge\; X_{i+1} = (X^+_i - W^{s^{\max}_i}_{M(A_i)}), \quad (14)$$

which is clearly contradictory.

To show that $(B^*_{i+1} \cup X_{i+1}) \subseteq \mathit{content}(t[i+1])$:

$$\begin{aligned}
& B^*_{i+1} \cup X_{i+1} && \\
&= (W^{s^{\min}_{i+1}}_{M(A_{i+1})} \cup X_{i+1}) && \{\text{because } k_{i+1} = 1\} \\
&= (W^{s^{\max}_i}_{M(A_{i+1})} \cup X_{i+1}) && \{\text{ because } s^{\min}_{i+1} = s^{\max}_i \text{ (shown in} \\
& && \quad \text{(13) and (14))}\} \\
&= (W^{s^{\max}_i}_{M(A_i)} \cup X_{i+1}) && \{\text{because } A_{i+1} = A_i\} \qquad (15) \\
&= \left(W^{s^{\max}_i}_{M(A_i)} \cup (X^+_i - W^{s^{\max}_i}_{M(A_i)})\right) && \{\text{because } X_{i+1} = (X^+_i - W^{s^{\max}_i}_{M(A_i)})\} \\
&= (W^{s^{\max}_i}_{M(A_i)} \cup X^+_i) && \{\text{immediate}\} \\
&\subseteq C_i && \{\text{by (12) and definition of } C_i\} \\
&= \mathit{content}(t[i+1]) && \{\text{by (a) for } i \text{ and Claim 1}\}.
\end{aligned}$$

To show that $\mathit{content}(t[i+1]) \subseteq (B^*_{i+1} \cup X_{i+1})$:

$$\begin{aligned}
\mathit{content}(t[i+1]) &= C_i && \{\text{by (a) for } i \text{ and Claim 1}\} \\
&= (B^*_i \cup X^+_i) && \{\text{by definition of } C_i\} \qquad (16) \\
&\subseteq (W^{s^{\max}_i}_{M(A_i)} \cup X^+_i) && \{\text{by (12)}\} \\
&= B^*_{i+1} \cup X_{i+1} && \{\text{by reasoning as in (15)}\}.
\end{aligned}$$

Finally, to show that $s^{\max}_i \in S_{i+1}$, the conditions in line 3 of Figure 2 are shown independently.

To show that $B^*_{i+1} \subseteq W^{s_i^{\max}}_{M(A_{i+1})} \subseteq C_{i+1}$:

$$\begin{aligned}
& B^*_{i+1} \\
&= W^{s_{i+1}^{\min}}_{M(A_{i+1})} && \{\text{because } k_{i+1} = 1\} \\
&= W^{s_i^{\max}}_{M(A_{i+1})} && \{\text{because } s_{i+1}^{\min} = s_i^{\max} \text{ (shown in (13) and (14))}\} \\
&= W^{s_i^{\max}}_{M(A_i)} && \{\text{because } A_{i+1} = A_i\} \\
&\subseteq C_i && \{\text{by (12)}\} \\
&= \mathit{content}(t[i+1]) && \{\text{by (a) for } i \text{ and Claim 1}\} \\
&\subseteq \mathit{content}(t[i+2]) && \{\text{immediate}\} \\
&= C_{i+1} && \{\text{ by (a) for } i+1 \text{ (shown in (15) and (16)) and} \\
& && \quad \text{Claim 1}\}.
\end{aligned} \tag{17}$$

To show that $s_i^{\max} \leq \max(C_{i+1})$:

$$\begin{aligned}
s_i^{\max} &\leq \max(C_i) && \{\text{by (12)}\} \\
&\leq \max(C_{i+1}) && \{\text{by reasoning as in (17)}\}.
\end{aligned} \tag{18}$$

To show that $\left(W^{s_i^{\max}+1}_{M(A_{i+1})} \cap (X^+_{i+1} - W^{s_i^{\max}}_{M(A_{i+1})})\right) \neq \emptyset$, note that $s_{i+1}^{\min} = s_i^{\max}$ (shown in (13) and (14)) and

$$(W^{s_{i+1}^{\min}}_{M(A_{i+1})} \cap X_{i+1}) = \emptyset \ \wedge \ (W^{s_{i+1}^{\min}+1}_{M(A_{i+1})} \cap X_{i+1}) \neq \emptyset \ \wedge \ X_{i+1} \subseteq X^+_{i+1}. \tag{19}$$

Case (III) $[\neg(\mathrm{I}) \ \wedge \ \neg(\mathrm{II})]$. Since $S_i = \emptyset$, by (b) for i, it must be the case that $k_i = 0$. Thus, $A_{i+1} = A_i$, $B_{i+1} = B_i$, $k_{i+1} = k_i$ $(= 0)$, and $X_{i+1} = X_i^+ \Big(= (X_i \cup \{x_i\})\Big)$ (line 9 of Figure 2). Furthermore,

$$\begin{aligned}
(B^*_{i+1} \cup X_{i+1}) &= (B_{i+1} \cup X_{i+1}) && \{\text{because } k_{i+1} = 0\} \\
&= (B_i \cup X_{i+1}) && \{\text{because } B_{i+1} = B_i\} \\
&= (B^*_i \cup X_{i+1}) && \{\text{because } k_i = 0\} \\
&= (B^*_i \cup X_i \cup \{x_i\}) && \{\text{because } X_{i+1} = (X_i \cup \{x_i\})\} \\
&= \Big(\mathit{content}(t[i]) \cup \{x_i\}\Big) && \{\text{by (a) for } i\} \\
&= \mathit{content}(t[i+1]) && \{\text{because } x \neq \#\}.
\end{aligned} \tag{20}$$

□ (*Claim 2*)

Claim 3. (a) through (e) below.

(a) $(\forall i)\Big[A_i \neq A_{i+1} \;\Rightarrow$
$\Big[B_i \subseteq A_{i+1} \subseteq \mathit{content}(t[i+1]) \;\wedge\; B_{i+1} = \mathit{content}(t[i+1])\Big]\Big]$.

(b) $(\forall i)[M(A_i)\!\downarrow]$.

(c) $(\forall i)[B_i \neq B_{i+1} \;\Rightarrow\; A_i \neq A_{i+1}]$.

(d) $(\forall i)\big[[k_i = 1 \;\wedge\; k_{i+1} = 0] \;\Rightarrow\; A_i \neq A_{i+1}\big]$.

(e) $(\forall i, j)\big[i \leq j \;\Rightarrow\; B_i \subseteq B_j \subseteq \mathit{content}(t[j])\big]$.

Proof of Claim. (a) follows from Claims 1 and 2(a), and from the construction of M'. (b) is shown by a straightforward induction. (c) and (d) are clear by the construction of M'. (e) follows from (a) and (c). □ (*Claim 3*)

Claim 4. There exists $i \in \mathbb{N}$ such that $(\forall j \geq i)[A_j = A_i]$.

Proof of Claim. By way of contradiction, suppose otherwise. By Lemma 22, there exists i_0 such that

$$(\forall \text{ finite } A')\Big[\mathit{content}(t[i_0]) \subseteq A' \subseteq L \;\Rightarrow\; M(A')\!\downarrow = M\Big(\mathit{content}(t[i_0])\Big)\Big]. \tag{21}$$

By Claim 3(a) and the (supposed) failure of the present claim, there exists $i_1 \geq i_0$ such that

$$\mathit{content}(t[i_0]) \subseteq B_{i_1}. \tag{22}$$

By Claim 3(a) and a second application of the failure of the present claim, there exists $i_2 > i_1$ such that

$$B_{i_1} \subseteq A_{i_2} \subseteq \mathit{content}(t[i_2]). \tag{23}$$

Note that, by (21) through (23),

$$M(A_{i_2})\!\downarrow = M\Big(\mathit{content}(t[i_0])\Big). \tag{24}$$

By a third application of the failure of the present claim, there exists $i_3 \geq i_2$ such that

$$A_{i_3} = A_{i_2} \;\wedge\; A_{i_3+1} \neq A_{i_2}. \tag{25}$$

By the construction of M' (i.e., line 4 of Figure 2), there must exist A' such that

$$B_{i_3} \subseteq A' \subseteq C_{i_3} \;\wedge\; M(A')\!\downarrow \neq M(A_{i_3})\!\downarrow. \tag{26}$$

Thus,

$$\begin{aligned}
content(t[i_0]) &\subseteq B_{i_1} && \{\text{by (22)}\} \\
&\subseteq B_{i_3} && \{\text{by Claim 3(e)}\} \\
&\subseteq A' && \{\text{by (26)}\} \\
&\subseteq C_{i_3} && \{\text{by (26)}\} \\
&= content(t[i_3+1]) && \{\text{by Claims 1 and 2(a)}\} \\
&\subseteq L && \{\text{immediate}\}.
\end{aligned} \tag{27}$$

Furthermore,

$$\begin{aligned}
M(A')\downarrow &\neq M(A_{i_3})\downarrow && \{\text{by (26)}\} \\
&= M(A_{i_2}) && \{\text{by (25)}\} \\
&= M\Big(content(t[i_0])\Big) && \{\text{by (24)}\}.
\end{aligned} \tag{28}$$

But this contradicts (21). □ (*Claim 4*)

Claim 5. There exists $i \in \mathbb{N}$ such that $(\forall j \geq i)[A_j = A_i \ \wedge \ B_j = B_i]$.

Proof of Claim. Immediate by Claims 3(c) and 4. □ (*Claim 5*)

Claim 6. There exists $i \in \mathbb{N}$ such that $W_{M(A_i)} = L$ and $(\forall j \geq i)[A_j = A_i \ \wedge \ B_j = B_i]$.

Proof of Claim. By Claim 5, there exists $i \in \mathbb{N}$ such that $(\forall j \geq i)[A_j = A_i \ \wedge \ B_j = B_i]$. By Claim 3(b), $M(A_i)\downarrow$. Clearly, by the construction of M', the condition in line 4 of Figure 2 *never* applies as M' is fed $x_i, x_{i+1}, \ldots$. Thus,

$$(\forall j \geq i)(\forall \text{ finite } A')\Big[[B_i \subseteq A' \subseteq C_j \ \wedge \ M(A')\downarrow] \ \Rightarrow \ M(A') = M(A_i)\Big]. \tag{29}$$

By Claim 3(e), $B_i \subseteq content(t[i])$ and, by Claims 1 and 2(a), for each $j \geq i$, $C_j = content(t[j+1])$. Thus,

$$(\forall j \geq i)\Big[M\Big(content(t[j])\Big)\downarrow \ \Rightarrow \ M\Big(content(t[j])\Big) = M(A_i)\Big]. \tag{30}$$

Clearly, then, $W_{M(A_i)} = L$. □ (*Claim 6*)

Claim 7. There exists $i \in \mathbb{N}$ such that $W_{M(A_i)} = L$ and $(\forall j \geq i)[A_j = A_i \ \wedge \ B_j = B_i \ \wedge \ k_j = 1]$.

Proof of Claim. By way of contradiction, suppose otherwise. By Claims 6, there exists i_0 such that $W_{M(A_{i_0})} = L$ and $(\forall j \geq i_0)[A_j = A_{i_0} \ \wedge \ B_j = B_{i_0}]$. By Claim 3(d), it must be the case that $(\forall j \geq i_0)[k_j = 0]$. By Claim 3(e), $B_{i_0} \subseteq \mathit{content}(t[i_0]) \subseteq L = W_{M(A_{i_0})}$. Thus, since L is infinite, there exists s_0 such that

$$B_{i_0} \subseteq W^{s_0}_{M(A_{i_0})} \subset W^{s_0+1}_{M(A_{i_0})}. \tag{31}$$

Again, since L is infinite, there exists $i_1 \geq i_0$ such that

$$\begin{aligned} & s_0 \leq \max\Big(\mathit{content}(t[i_1+1])\Big) \\ \wedge\ & W^{s_0}_{M(A_{i_0})} \subseteq \mathit{content}(t[i_1+1]) \\ \wedge\ & \Big(W^{s_0+1}_{M(A_{i_0})} \cap \Big(\mathit{content}(t[i_1+1]) - W^{s_0}_{M(A_{i_0})}\Big)\Big) \neq \emptyset. \end{aligned} \tag{32}$$

By (32) and Claims 1 and 2(a),

$$\begin{aligned} & s_0 \leq \max(C_{i_1}) \\ \wedge\ & W^{s_0}_{M(A_{i_0})} \subseteq C_{i_1} \\ \wedge\ & \Big(W^{s_0+1}_{M(A_{i_0})} \cap (C_{i_1} - W^{s_0}_{M(A_{i_0})})\Big) \neq \emptyset. \end{aligned} \tag{33}$$

Note that $C_{i_1} = B^*_{i_1} \cup X^+_{i_1}$ and

$$\begin{aligned} B^*_{i_1} &= B_{i_1} && \{\text{because } k_{i_1} = 0\} \\ &= B_{i_0} && \{\text{because } B_{i_1} = B_{i_0}\} \\ &\subseteq W^{s_0}_{M(A_{i_0})} && \{\text{by (31)}\}. \end{aligned} \tag{34}$$

By (33) and (34), it must be the case that

$$\Big(W^{s_0+1}_{M(A_{i_0})} \cap (X^+_{i_1} - W^{s_0}_{M(A_{i_0})})\Big) \neq \emptyset. \tag{35}$$

By (33) through (35), and the fact that $A_{i_0} = A_{i_1}$,

$$\begin{aligned} & s_0 \leq \max(C_{i_1}) \\ \wedge\ & B^*_{i_1} \subseteq W^{s_0}_{M(A_{i_1})} \subseteq C_{i_1} \\ \wedge\ & \Big(W^{s_0+1}_{M(A_{i_1})} \cap (X^+_{i_1} - W^{s_0}_{M(A_{i_1})})\Big) \neq \emptyset. \end{aligned} \tag{36}$$

Thus, by the construction of M', $s_0 \in S_{i_1}$ and $k_{i_1+1} = 1$ (a contradiction).
□ (*Claim 7*)

Claim 8. Let i_0 be the i asserted to exist by Claim 7. Then, for each $s \in \mathbb{N}$, there exists $j \geq i_0$ such that $s \leq s^{\min}_j$.

Proof of Claim. By way of contradiction, let $s_0 \in \mathbb{N}$ be such that, for each $j \geq i_0$, $s_j^{\min} < s_0$. By Claim 3(e), $B_{i_0} \subseteq \mathit{content}(t[i_0]) \subseteq L = W_{M(A_{i_0})}$. Thus, since L is infinite, there exists $s_1 \geq s_0$ such that

$$B_{i_0} \subseteq W_{M(A_{i_0})}^{s_1} \subset W_{M(A_{i_0})}^{s_1+1}. \tag{37}$$

Again, since L is infinite, there exists $i_1 \geq i_0$ such that

$$\begin{aligned} & s_1 \leq \max\Big(\mathit{content}(t[i_1+1])\Big) \\ \wedge\ & W_{M(A_{i_0})}^{s_1} \subseteq \mathit{content}(t[i_1+1]) \\ \wedge\ & \Big(W_{M(A_{i_0})}^{s_1+1} \cap \Big(\mathit{content}(t[i_1+1]) - W_{M(A_{i_0})}^{s_1}\Big)\Big) \neq \emptyset. \end{aligned} \tag{38}$$

By (38) and Claims 1 and 2(a),

$$\begin{aligned} & s_1 \leq \max(C_{i_1}) \\ \wedge\ & W_{M(A_{i_0})}^{s_1} \subseteq C_{i_1} \\ \wedge\ & \Big(W_{M(A_{i_0})}^{s_1+1} \cap (C_{i_1} - W_{M(A_{i_0})}^{s_1})\Big) \neq \emptyset. \end{aligned} \tag{39}$$

Note that $C_{i_1} = B_{i_1}^* \cup X_{i_1}^+$ and

$$\begin{aligned} B_{i_1}^* &= W_{M(A_{i_1})}^{s_{i_1}^{\min}} \ \{\text{because } k_{i_1} = 1\} \\ &\subseteq W_{M(A_{i_1})}^{s_1} \ \{\text{because } s_{i_1}^{\min} < s_0 \leq s_1\} \\ &= W_{M(A_{i_0})}^{s_1} \ \{\text{because } A_{i_1} = A_{i_0}\}. \end{aligned} \tag{40}$$

By (39) and (40), it must be the case that

$$\Big(W_{M(A_{i_0})}^{s_1+1} \cap (X_{i_1}^+ - W_{M(A_{i_0})}^{s_1})\Big) \neq \emptyset. \tag{41}$$

By (39) through (41), and the fact that $A_{i_0} = A_{i_1}$,

$$\begin{aligned} & s_1 \leq \max(C_{i_1}) \\ \wedge\ & B_{i_1}^* \subseteq W_{M(A_{i_1})}^{s_1} \subseteq C_{i_1} \\ \wedge\ & \Big(W_{M(A_{i_1})}^{s_1+1} \cap (X_{i_1}^+ - W_{M(A_{i_1})}^{s_1})\Big) \neq \emptyset. \end{aligned} \tag{42}$$

Thus, by the construction of M', $s_1 \in S_{i_1}$ and $s_1 \leq s_{i_1}^{\max}$. Finally, by Claim 2(b), $s_1 \leq s_{i_1+1}^{\min}$ (a contradiction). □ (*Claim 8*)

Claim 9. For each $i \in \mathbb{N}$, there exists $j \geq i$ such that $x_i \notin X_{j+1}$.

Proof of Claim. Follows from Claim 8. □ (*Claim 9*)

□ (*Theorem 23*)

Corollary 24 (of Corollary 21 and Theorem 23) Let $\mathcal{L}$ be any class of infinite languages. Then, $\mathcal{L} \in \mathit{LimTxt}$ iff $\mathcal{L} \in \mathit{Tem}_*\mathit{Txt}$.

For most of the remaining separation results of Section 3, it is currently open whether that separation can be witnessed by classes of infinite languages. However, as Theorem 26 below shows, any class of infinite languages that can be Bem_1-learned can be Tem_2-learned. Intuitively, this says that, for Bem_1-learnable classes of infinite languages, restriction to temporary memory is recouped by being able to store just 1 additional example.

The proof of the aforementioned result requires the following technical lemma.

Lemma 25 There exists a computable function $f : \mathbb{N}^2 \to \mathbb{N}$ such that

$$(\forall p)\Big[W_p \text{ is infinite } \Rightarrow (\forall s)(\exists x \in W_p)[f(p,x) = s]\Big]. \tag{43}$$

Proof. For each $p \in \mathbb{N}$, $f(p,\cdot)$ is constructed in stages. Let $f(p,0) = 0$ and $x^0 = 0$. Go to stage 0.

Begin stage s.
0. For $y = x^s, x^s+1, ...$: if $W_p^y \cap \{x^s, x^s+1, ..., y\} = \emptyset$, then set $f(p, y+1) = f(p, x^s)$; otherwise, proceed to the next step.
1. Let $y \geq x^s$ be *least* such that $W_p^y \cap \{x^s, x^s+1, ..., y\} \neq \emptyset$.
2. Set $f(p, y+1) = s+1$.
3. Set $x^{s+1} = y+1$.
4. Proceed to stage $s+1$.
End stage s.

Clearly, f is computable. To show that f satisfies (43), let $p \in \mathbb{N}$ be such that W_p is infinite, and let $s \in \mathbb{N}$ be fixed. It is straightforward to show that $f(p, x^s) = s$. Furthermore, since W_p is infinite, there exists a (least) $y \geq x^s$ such that $W_p^y \cap \{x^s, x^s+1, ..., y\} \neq \emptyset$. Choose $x \in W_p^y \cap \{x^s, x^s+1, ..., y\}$ arbitrarily. Clearly, by the construction of f, $f(p,x) = f(p,x^s) = s$. □ (*Lemma 25*)

Theorem 26 For each class of infinite languages $\mathcal{L}$, if $\mathcal{L} \in \mathit{Bem}_1\mathit{Txt}$, then $\mathcal{L} \in \mathit{Tem}_2\mathit{Txt}$.

Proof. Let $\mathcal{L}$ be a class of infinite languages such that $\mathcal{L} \in \mathit{Bem}_1\mathit{Txt}$, and let (M, p_0) be such that (M, p_0) $\mathit{Bem}_1\mathit{Txt}$-identifies $\mathcal{L}$. An M' is constructed such that (M', p_0) $\mathit{Tem}_2\mathit{Txt}$-identifies $\mathcal{L}$. The conjectures of M' will be of the form

M'		M	
conjecture	memory	conjecture	memory
$\langle p, 0\rangle$	$\emptyset$	p	$\emptyset$
$\langle p, u+1\rangle$	$\emptyset$	p	$\{u\}$
$\langle p, 0\rangle$	$\{u\}$	p	$\{u\}$
$\langle p, 0\rangle$	$\{u, v\}$ $(u \neq v)$	p	$(\pi_1^2 \circ f)(p, \max\{u, v\})$

Fig. 3. The correspondence between the conjectures/memory of M', and those of M, on any initial segment of text.

$\langle p, k\rangle$, where p is a conjecture of M, and k is either 0, or $u+1$ for some element u appearing in the input text. Let $f : \mathbb{N}^2 \to \mathbb{N}$ be a computable function as asserted to exist by Lemma 25.

For this proof only, let $\pi_1^2 : \mathbb{N} \to \mathbb{N}$ be such that, for each $x, y \in \mathbb{N}$, $\pi_1^2(\langle x, y\rangle) = x$. Note that such a function exists, and is computable, since $\langle \cdot, \cdot \rangle$ is onto and computable. Further note that, for each $p \in \mathbb{N}$, if W_p is infinite, then, for each $u \in \mathbb{N}$, there exist *infinitely many* $x \in W_p$ such that $(\pi_1^2 \circ f)(p, x) = u$.

Figure 3 gives the correspondence between the conjectures/memory of M', and those of M, on any initial segment of text. M' operates by simulating M in a manner which respects the invariants given in the figure.

Suppose that $(\langle p, k\rangle, A')$ are the current conjecture/memory of M', and that (p, A) are the corresponding conjecture/memory of M (as per Figure 3). Further suppose that x is the next element appearing in the input text. If $M(p, A, x)\uparrow$, then $M'(\langle p, k\rangle, A', x)\uparrow$. Otherwise, let $(q, B) = M(p, A, x)$. Then, on input x, M' outputs the following.

$$\begin{cases} (\langle q, 0\rangle, \emptyset), & \text{if } [p \neq q \ \wedge \ B = \emptyset]; \\ (\langle q, u+1\rangle, \emptyset), & \text{if } [p \neq q \ \wedge \ B \neq \emptyset], \text{ where } B = \{u\}; \\ (\langle p, 0\rangle, B), & \text{if } [p = q \ \wedge \ A \neq B]; \\ (\langle p, 0\rangle, \{\max A', x\}), & \text{if } \Big[p = q \ \wedge \ A = B \ \wedge \ A' \neq \emptyset \ \wedge \ \max A' < x \\ & \quad \wedge \ [|A'| = 1 \ \Rightarrow \ (\pi_1^2 \circ f)(p, x) = \max A'] \\ & \quad \wedge \ [|A'| = 2 \ \Rightarrow \ (\pi_1^2 \circ f)(p, x) = (\pi_1^2 \circ f)(p, \max A')]\Big]; \\ (\langle p, 0\rangle, A'), & \text{otherwise.} \end{cases} \tag{44}$$

It is straightforward to verify that M' respects the invariants of Figure 3. Similarly, the following facts are straightforward to verify.

- If the conjecture of M converges to some p, then the conjecture of M' converges to p.
- If the memory of M converges to $\emptyset$, then the memory of M' converges to $\emptyset$.
- If the memory of M changes infinitely often, then M' satisfies the temporary memory requirement.

All that remains to be shown is that: if, on some text for a language in $\mathcal{L}$, the memory of M converges to a singleton set, then M' satisfies the temporary memory requirement.

Suppose that, on some text for a language in $\mathcal{L}$, M converges to $(p, \{u\})$. Then, clearly, either M' will converge to $(\langle p, u+1\rangle, \emptyset)$, or there will be some point at which M' outputs $(\langle p, 0\rangle, \{u\})$. In the former case, M' satisfies the temporary memory requirement. So, suppose the latter case. Note that since conjecture p correctly identifies the contents of the input text, W_p is infinite. Thus, eventually, there will appear some x in the text such that $u < x$ and $(\pi_1^2 \circ f)(p, x) = u$. Upon seeing this x, M' will change its memory to $\{u, x\}$.

Suppose that, at some subsequent point, the memory of M' is $\{v, w\}$, where $v \neq w$. Without loss of generality, suppose that $v < w$. Then, eventually, there will appear some x in the text such that $w < x$ and $(\pi_1^2 \circ f)(p, x) = (\pi_1^2 \circ f)(p, w)$. Upon seeing this x, M' will change its memory to $\{w, x\}$.

Thus, M' satisfies the temporary memory requirement. □ (*Theorem 26*)

Problem 27 (a) Is there a class of infinite languages $\mathcal{L} \in \mathit{Bem}_1\mathit{Txt} - \mathit{Tem}_1\mathit{Txt}$?
(b) Let $k \geq 2$, $\mathfrak{A} \in \{\mathit{Bem}_2\mathit{Txt}, ..., \mathit{Bem}_k\mathit{Txt}\}$, and $\mathfrak{B} \in \{\mathit{Tem}_k\mathit{Txt}, \mathit{Tem}_{k+1}\mathit{Txt}, ..., \mathit{Tem}_*\mathit{Txt}\}$. Is there a class of infinite languages $\mathcal{L} \in \mathfrak{A} - \mathfrak{B}$?

5 Conclusion

We introduced a new model of language learning called *temporary example memory* (*Tem*) learning. This model is a natural *restriction* of bounded example memory (*Bem*) learning. In particular, it requires that, if a learner commits an example x to memory in some stage of the learning process, then there is some subsequent stage for which x *no longer* appears in the learner's memory. In some sense, this model captures the idea that *memories fade*.

Aside from the open question mentioned in Section 4, the following would constitute an interesting line of research. In some sense, an IIM can memorize examples that it has seen by *coding* them into its hypotheses, i.e., by exploiting redundancy in the hypothesis space. This "memory" is, in principle, unbounded in the number of examples that it can retain, and in how long it can retain them.[7] From a practical point of view, the option to memorize examples in this way probably does not meet the *intuitive* requirements of a model of incremental learning. Thus, it would be interesting to consider the *Bem* and *Tem*-learning models in conjunction with hypothesis spaces *that have no redundancy*, i.e., Friedberg numberings. Note that such numberings have already been considered as hypothesis spaces in the context of *It*-learning [JS08].

Acknowledgements. Thanks are due to our anonymous reviewers, for many insightful comments and helpful suggestions.

References

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

[Bak02] B. Bakker. Reinforcement learning with long short-term memory. *Advances in Neural Information Processing Systems*, 14:1475–1482, 2002.

[BB75] L. Blum and M. Blum. Toward a mathematical theory of inductive inference. *Information and Control*, 28(2):125–155, 1975.

[CCJS07] L. Carlucci, J. Case, S. Jain, and F. Stephan. Results on memory-limited U-shaped learning. *Information and Computation*, 205(10):1551–1573, 2007.

[CJLZ99] J. Case, S. Jain, S. Lange, and T. Zeugmann. Incremental concept learning for bounded data mining. *Information and Computation*, 152(1):74–110, 1999.

[Ful90] M. Fulk. Prudence and other conditions on formal language learning. *Information and Computation*, 85(1):1–11, 1990.

[Gol67] E.M. Gold. Language identification in the limit. *Information and Control*, 10(5):447–474, 1967.

[HMU01] J.E. Hopcroft, R. Motwani, and J.D. Ullman. *Introduction to automata theory, languages, and computation.* Addison Wesley, second edition, 2001.

[HS97] S. Hochreiter and J. Schmidhuber. Long short-term memory. *Neural Computation*, 9(8):1735–1780, 1997.

[JS08] S. Jain and F. Stephan. Learning in Friedberg numberings. *Information and Computation*, 206(6):776–790, 2008.

[7] Of course, since the IIM must eventually converge to a single hypothesis, the IIM can memorize examples in this way only finitely often.

[KS95] E. Kinber and F. Stephan. Language learning from texts: mind changes, limited memory, and monotonicity. *Information and Computation*, 123(2):224–241, 1995.

[LM92] L.J. Lin and T. Mitchell. Reinforcement learning with hidden states. In *Proc. of the 2nd International Conference on Simulation of Adaptive Behavior*, pages 271–280, 1992.

[LMZ08] S. Lange, S.E. Moelius, and S. Zilles. Learning with temporary memory (expanded version). In *Proc. of the 19th International Conference on Algorithmic Learning Theory*, volume 5254 of *Lecture Notes in Computer Science*, pages 449–463. Springer, 2008.

[LZ96] S. Lange and T. Zeugmann. Incremental learning from positive data. *Journal of Computer and System Sciences*, 53(1):88–103, 1996.

[LZZ08] S. Lange, T. Zeugmann, and S. Zilles. Learning indexed families of recursive languages from positive data: A survey. *Theoretical Computer Science*, 397(1-3):194–232, 2008.

[McC96] R.A. McCallum. Learning to use selective attention and short-term memory in sequential tasks. In *Proc. of the 4th International Conference on Simulation of Adaptive Behavior*, pages 315–324, 1996.

[Mit97] T.M. Mitchell. *Machine Learning*. McGraw-Hill Higher Education, 1997.

[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., first edition, 1986.

[Rog67] H. Rogers. *Theory of Recursive Functions and Effective Computability*. McGraw Hill, New York, 1967. Reprinted, MIT Press, 1987.

[WC80] K. Wexler and P.W. Culicover. *Formal Principles of Language Acquisition*. MIT Press, Cambridge, Mass., 1980.

[Wie76] R. Wiehagen. Limes-Erkennung rekursiver Funktionen durch spezielle Strategien. *Elektronische Informationsverarbeitung und Kybernetik*, 12(1/2):93–99, 1976.