Computability 13 (2024) 237–261
DOI 10.3233/COM-220421
IOS Press

Maps for learning indexable classes

Julian Berger
Hasso Plattner Institute, University of Potsdam, Germany
Julian.Berger@student.hpi.uni-potsdam.de

Maximilian Böther
Hasso Plattner Institute, University of Potsdam, Germany
Maximilian.Boether@student.hpi.uni-potsdam.de

Vanja Doskoč
Hasso Plattner Institute, University of Potsdam, Germany
Vanja.Doskoc@hpi.de

Jonathan Gadea Harder
Hasso Plattner Institute, University of Potsdam, Germany
Jonathan.GadeaHarder@student.hpi.uni-potsdam.de

Nicolas Klodt
Hasso Plattner Institute, University of Potsdam, Germany
Nicolas.Klodt@hpi.de

Timo Kötzing
Hasso Plattner Institute, University of Potsdam, Germany
Timo.Koetzing@hpi.de

Winfried Lötzsch
Hasso Plattner Institute, University of Potsdam, Germany
Winfried.Loetzsch@student.hpi.uni-potsdam.de

Jannik Peters
Hasso Plattner Institute, University of Potsdam, Germany
Jannik.Peters@student.hpi.uni-potsdam.de

Leon Schiller
Hasso Plattner Institute, University of Potsdam, Germany
Leon.Schiller@student.hpi.uni-potsdam.de

Lars Seifert
Hasso Plattner Institute, University of Potsdam, Germany
Lars.Seifert@student.hpi.uni-potsdam.de

Armin Wells
Hasso Plattner Institute, University of Potsdam, Germany
Armin.Wells@student.hpi.uni-potsdam.de

Simon Wietheger
Hasso Plattner Institute, University of Potsdam, Germany
Simon.Wietheger@student.hpi.uni-potsdam.de

Abstract. We study learning of indexed families from positive data where a learner can freely choose a hypothesis space (with uniformly decidable membership) comprising at least the languages to be learned. This abstracts a very universal learning task which can be found in many areas, for example learning of (subsets of) regular languages or learning of natural languages. We are interested in various restrictions on learning, such as consistency, conservativeness or set-drivenness, exemplifying various natural learning restrictions.

The contribution of this work is twofold. First, we present a general result on how the hypothesis spaces may be constructed *during* learning, rather than beforehand. Using this result, we build on previous results from the literature and provide several maps (depictions of all pairwise relations) of various groups of learning criteria, including a map for monotonicity restrictions and similar criteria and a map for restrictions on data presentation. Furthermore, we consider, for various learning criteria, whether learners can be assumed consistent.

Keywords: Language learning in the limit, indexed family, hypothesis space, map, characteristic index

1. Introduction

We investigate the problem of algorithmically learning a description for a formal language (a computably enumerable subset of the set of all natural numbers) when presented successively all and only the elements of that language. This is called *inductive inference*, a branch of (algorithmic) learning theory. For example, a learner h (a computable device) might be presented more and more numbers. After each new number, h outputs a description for a language as its conjecture. The learner h might decide to output a program for the set of all odd numbers, as long as all numbers presented are odd and prime. Later, when h sees a 2, it might change this guess to a program for the set of all prime numbers.

In the literature, many criteria for determining whether a learner h is *successful* on a (target) language L have been proposed. Gold [15] gave a first, simple learning criterion, **TxtGEx**-*learning*,[1] where a learner is *successful* if and only if, on every *text* for L (listing of all and only the elements of L) it eventually stops changing its conjectures, and its final conjecture is a correct description for the input language. Trivially, each single, describable language L has a suitable constant function as a **TxtGEx**-learner (this learner constantly outputs a description for L). Thus, we are interested in analyzing for which *classes of languages* $\mathcal{L}$ there is a *single learner* h learning *each* member of $\mathcal{L}$. This framework is also known as *language learning in the limit* and has been studied extensively, using a wide range of learning criteria similar to **TxtGEx**-learning (see, for example, the textbook [16]).

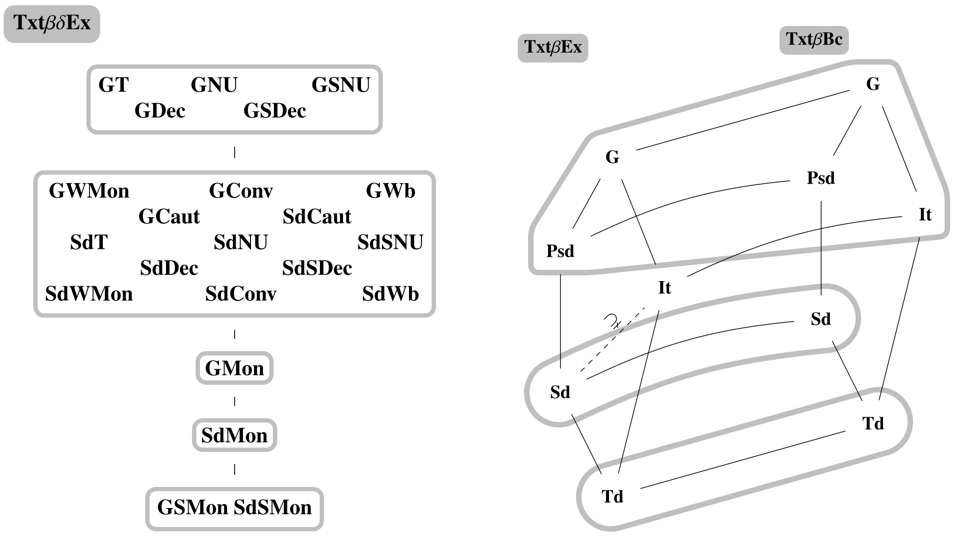
A major branch of this analysis focuses on learning *indexed families*, that is, classes of languages $\mathcal{L}$ such that there is an enumeration $(L_i)_{i\in\mathbb{N}}$ of all and only the elements of $\mathcal{L}$ for which the decision problem "$x \in L_i$" is decidable. Already for such classes of languages we get a rich structure, see the survey of previous work [29]. Here, the learners learn with respect to some indexed family, the so-called *hypothesis space*. The conjectures of the learners are then interpreted according to the hypothesis space and the learners learn a class of languages successfully if they do so with respect to some hypothesis space. We are specifically interested in *class comprising* learning, where our learners are free to choose any hypothesis space containing hypotheses at least for the target languages. This is in contrast, for example, to learning with an explicitly given hypothesis space.

Since the appearance of the mentioned survey, only little work on indexable classes was conducted, while learning of arbitrary families of languages sprouted a new mode of analysis, *map charting*. This approach tries to further the understanding of learning settings by looking at all relations between similar learning criteria and displaying them as a *map* [21,22]. This approach builds on the pairwise relations which are already known in the literature and completes them in interesting settings to understand one aspect more closely, for example regarding certain natural restrictions on what kind of mind changes are allowed (which we will consider in Section 5) or the importance of data presentation (which we will consider in Section 6).

We develop a very useful characterization of learning indexable families given in Theorem 1. Here we show that learnability of an indexed family with an arbitrary hypothesis space is equivalent to the learnability by a learner which only outputs programs for characteristic functions [4] and is considered successful when converging to such a program which decides the target language. This result allows us to simplify many of our proofs, since now the hypothesis space does not need to be chosen in advance.

With this characterization, we start our analysis by considering the restriction of *consistent* learning [1]. A learner is consistent if and only if each of its hypotheses correctly reflects the data which the hypothesis is based on. Note that, for arbitrary learning in the Gold-style model, learners cannot be assumed consistent in general [3], a result termed the *inconsistency phenomenon*. The reason behind this result is essentially the same as for the halting problem: a general hypothesis cannot be checked for consistency in a computable way. Since, for indexed families, consistency of hypotheses is decidable, it comes at no surprise that here learners can, in general, be assumed

[1] **Txt** stands for learning from a *text* of positive examples; **G** stands for *Gold-style* learning and indicates that the learner has full information on the data given; **Ex** stands for *explanatory*.



(a) (b)

Figure 1. Depiction of relations between **G**- and **Sd**-learners under various additional restrictions (see Fig. 1(a)) and syntactic and semantic convergence criteria under various memory restrictions β for total learners (see Fig. 1(b)) when learning indexable classes. For a full list of the considered restrictions, please see Section 2. Black solid lines imply inclusions (bottom-to-top, left-to-right). In Fig. 1(b), these inclusions are trivial and the dashed line depicts the non-trivial proper inclusion $[\mathcal{R}\mathbf{TxtItEx}]_{\mathbf{ind}} \subsetneq [\mathcal{R}\mathbf{TxtSdEx}]_{\mathbf{ind}}$. Greyly edged areas illustrate a collapse of the enclosed learning criteria. There are no further collapses.

consistent [37]. However, to prove this result, crucial changes to the hypotheses are made (so as to make them consistent), which might spoil other nice properties the learner might exhibit (such as never overgeneralizing the true target language). In Section 4 we show several different ways in which total learners can be made consistent, each maintaining other restrictions (such as, for example, *conservative* learning [1], where learners must not change their mind while their hypothesis is consistent).

In Section 5 we consider one of the best-studied maps from other learning settings, the map of *delayable* learning restrictions. We build on previously known results, such as that conservative learning is restrictive [24], and complete the map both for the case of *full information* (where the learner has access to the full history of data shown) and for *set-driven* learners (which only have access to the set of data presented so far, but not to the order of presentation [34]). This builds on earlier analyses of monotone learning which has been studied in various settings [27]. We depict our results in Fig. 1(a). In particular, we show that the criteria cluster into merely five different learning powers, i.e., many learning criteria allow for learning the same classes of languages. Among other things, we show that *(strong) non-U-shaped* learning [2,11], where abandoning a correct hypothesis is forbidden, is not restrictive in either setting (set-driven and full-information).

In Section 6 we consider in more detail what impact the access to information has on the learning power of total learners. Additionally to full information and set-driven learning, we also consider *iterative* learning (where the learner has access to its previous hypothesis, but only the current datum [34]). We give the complete map for **Ex**-learning at the same time as **Bc**-learning (*behaviorally correct* learning, where the learner need not to stop syntactically changing the conjecture, as long as it remains semantically correct [9,31]). We depict our findings in Fig. 1(b).

These directions taken together (consistency, delayable restrictions, information access, syntactic vs. semantic convergence) give a well-rounded picture, offering a glimpse on learning indexable classes from all commonly studied angles.

2. Preliminaries

In this section we introduce the mathematical notations and notions used throughout the paper. For non-introduced notation we refer to [32]. Regarding the learning criteria, we follow the system of [19].

2.1. Language learning in the limit

We denote the set of natural numbers by $\mathbb{N} = \{0, 1, 2, \ldots\}$. With $\subseteq$ and $\subsetneq$ we denote the subset and proper subset relation between sets, respectively. With $\cap, \cup, \setminus$ we denote the set intersection, union, and difference, respectively. We let $\emptyset$ and ε denote the empty set and empty sequence, respectively. We use $\mathcal{P}$ ($\mathcal{R}$) as the set of all (total) computable functions. If a function f is defined on an argument x we denote this by $f(x)\downarrow$; otherwise, we write $f(x)\uparrow$. We fix an effective numbering $(\varphi_e)_{e\in\mathbb{N}}$ of $\mathcal{P}$, where e may be viewed as a *program* or *index* for the function φ_e.

We fix the symbol # called *pause*. For any set $S \subseteq \mathbb{N}$, we denote $S_\# := S \cup \{\#\}$. The set of all sequences of length at most $t \in \mathbb{N}$ over $S_\#$ is denoted by $S_\#^{\leqslant t}$ and the set of all finite sequences over $\mathbb{N} \cup \{\#\}$ by $\mathbb{S}\mathrm{eq}$. For two sequences σ, τ, we let $\sigma^\frown\tau$ denote their concatenation and we write $\sigma \subseteq \tau$ ($\sigma \subsetneq \tau$) if and only if σ is a (proper) prefix of τ. For a (possibly infinite) sequence σ, we let $\mathrm{content}(\sigma) = \mathrm{range}(\sigma) \setminus \{\#\}$. For $\sigma \in \mathbb{S}\mathrm{eq}$, we denote the sequence with the last element removed as σ^-. Furthermore, we may interpret finite sequences as natural numbers and fix a total order $\leqslant$ on these such that, in particular, for all $\sigma, \tau \in \mathbb{S}\mathrm{eq}$ with $\sigma \subseteq \tau$ we have that $\sigma \leqslant \tau$.

We call a computably enumerable set $L \subseteq \mathbb{N}$ a *language*. We learn *indexed families* of languages, that is, families of languages $(L_i)_{i\in\mathbb{N}}$ where there exists a total computable function f such that, for all $i, x \in \mathbb{N}$,

$$f(i, x) = \begin{cases} 1, & \text{if } x \in L_i; \\ 0, & \text{otherwise.} \end{cases}$$

We learn these families with respect to hypothesis spaces, which are indexed families of languages themselves. In general, a *learner* is a function $h \in \mathcal{P}$. We examine learning from text. A *text* is a total function $T : \mathbb{N} \to \mathbb{N} \cup \{\#\}$. We denote the set of all texts as **Txt**. Similar to sequences, we define the *content* of a text $T \in \mathbf{Txt}$ as $\mathrm{content}(T) = \mathrm{range}(T) \setminus \{\#\}$. Furthermore, we call T a *text for a language* L if $\mathrm{content}(T) = L$; the set of all texts for L is denoted by $\mathbf{Txt}(L)$. The *canonical text* of a language L is the enumeration of all elements in L in strictly ascending order (if L is finite, the text returns # after all elements have been presented). Analogously, the *canonical sequence* of a (finite) language L is the (finite) sequence of all elements in L in strictly ascending order. Additionally, we define $T[0] = \varepsilon$ and, for all $n \in \mathbb{N}$ with $n > 0$, $(T[n] = T(0), \ldots, T(n-1))$.

What kind of information a learner is given, is specified by an *interaction operator*. Formally, an interaction operator is a function that takes a learner and a text as input arguments and outputs a (possibly partial) function that is called *learning sequence* or *sequence of hypotheses*. We consider *Gold-style* or *full-information* learning [15], denoted by **G**, *iterative* learning (**It**, [13,35]), *partially set-driven* or *rearrangement-independent* learning (**Psd**, [5,33]), *set-driven* learning (**Sd**, [34]) and *transductive* learning (**Td**, [7,19]). Note that transductive learners may output a special symbol "?" if the information given is not sufficient to make a guess. Formally, for all learners $h \in \mathcal{P}$, texts $T \in \mathbf{Txt}$ and $i \in \mathbb{N}$,

$$\mathbf{G}(h, T)(i) = h\big(T[i]\big);$$

$$\mathbf{Psd}(h, T)(i) = h\big(\mathrm{content}\big(T[i]\big), i\big);$$

$$\mathbf{Sd}(h, T)(i) = h\big(\mathrm{content}\big(T[i]\big)\big);$$

$$\mathbf{It}(h, T)(i) = \begin{cases} h(\varepsilon), & \text{if } i = 0; \\ h(\mathbf{It}(h, T)(i-1), T(i-1)), & \text{otherwise;} \end{cases}$$

$$\mathbf{Td}(h, T)(i) = \begin{cases} ?, & \text{if } i = 0; \\ \mathbf{Td}(h, T)(i-1), & \text{else, if } h(T(i-1)) = ?; \\ h(T(i-1)), & \text{otherwise.} \end{cases}$$

Intuitively, Gold-style learners have full information on the elements given. Set-driven learners base their hypotheses solely on the content of the information given, while partially set-driven learners additionally have a counter for the iteration step. Iterative learners base their conjectures on their previous hypothesis and the current datum. Lastly, transductive learners solely base their guesses on the current datum and may output "?" if the information is not sufficient.

For two interaction operators β, β' we write $\beta \preceq \beta'$ if and only if every β-learner h can be compiled into an equivalent β'-learner h' such that, for any text T, we have $\beta(h, T) = \beta'(h', T)$. We note that $\mathbf{Td} \preceq \mathbf{It} \preceq \mathbf{G}$ and $\mathbf{Sd} \preceq \mathbf{Psd} \preceq \mathbf{G}$. As an example, every **Sd**-learner can be compiled into an **Psd**-learner by simply ignoring the counter. Furthermore, note that any **Td**-learner may be simulated by a **Sd**-learner, however, the order of the hypotheses may be changed. For any β-learner h with $\beta \preceq \mathbf{G}$, we let h^*, the *starred learner*, denote the G-learner simulating h. For example, the starred learner of a **Psd**-learner h is defined, for all sequences σ, as $h^*(\sigma) = h(\text{content}(\sigma), |\sigma|)$.

For a learner to successfully identify a language it has to satisfy certain restrictions. A famous example was given by Gold, who required the learner to converge to a correct hypothesis for the target language [15]. This is called *explanatory* learning and denoted by **Ex**. When we speak of correct hypotheses, it is with regard to an indexed hypothesis space. Formally, a learning restriction is a predicate on a sequence of hypotheses p and a text $T \in \mathbf{Txt}$. In the case of explanatory learning, we get, for a given indexed hypothesis space $\mathcal{H} = (H_i)_{i\in\mathbb{N}}$,

$$\mathbf{Ex}(p, T) \Leftrightarrow \exists n_0 \forall n \geqslant n_0\colon\ p(n) = p(n_0) \wedge H_{p(n_0)} = \text{content}(T).$$

We now give the intuition for the considered restrictions and define them formally afterwards. As an alternative to **Ex**, for *behaviorally-correct* (**Bc**) learning one only requires semantic convergence, that is, after some point all hypotheses must be correct hypotheses for the target language, but they do not need to be syntactically equal [9,31].

In addition to these convergence criteria there are various other properties to require from a learner that are inspired from nature or other sciences. In *non-U-shaped* learning (**NU**, [2]), once the learner outputs a correct hypothesis, it may not unlearn the language, i.e., it may only make syntactic mind changes. In *strongly non-U-shaped* learning (**SNU**, [11]) not even these syntactic mind changes are allowed. In *consistent* learning (**Cons**, [1]), each hypothesis must include the given information. There exist various monotonicity restrictions ([17,25,36]). When learning *strongly monotone* (**SMon**), the learner may not discard elements present in previous hypotheses, and in *monotone* learning (**Mon**) the learner is not allowed to remove correct data from its hypotheses. Furthermore, in *weakly monotone* learning (**WMon**) the learner must remain strongly monotone while consistent with the input. Similarly, in *cautious* learning (**Caut**, [30]), no hypothesis may be a proper subset of a prior hypothesis. As a relaxation, in *target-cautious* learning ($\mathbf{Caut}_{\mathbf{Tar}}$, [21]), no hypothesis may be a proper superset of the target language. A specialization of cautious and weakly monotone learning is *witness-based* learning (**Wb**, [22]), where each mind change must be justified by a witness. Witness-based learning is also a specialization of *conservative* learning (**Conv**, [1]) where the learner may only make a mind change when an inconsistency is detected. If we only require this for semantic mind changes, we call the learner *semantically conservative* (**SemConv**, [23]). Finally, in *decisive* learning (**Dec**, [30]), the learner may not return to semantically abandoned hypotheses; in *strongly decisive* learning (**SDec**, [20]), the learner may not return to syntactically abandoned hypotheses. Now, we give formal definitions for theses restrictions. Let $\mathcal{H} = (H_i)_{i\in\mathbb{N}}$ be an indexed hypothesis space. For any sequence of hypotheses p and text $T \in \mathbf{Txt}$, define

$$\mathbf{Bc}(p, T) \Leftrightarrow \exists n_0 \forall n \geqslant n_0\colon\ H_{p(n)} = \text{content}(T);$$

$$\mathbf{NU}(p, T) \Leftrightarrow \forall i, j, k\colon \big(i \leqslant j \leqslant k \wedge H_{p(i)} = H_{p(k)} = \text{content}(T)\big) \Rightarrow H_{p(i)} = H_{p(j)};$$

$$\mathbf{SNU}(p, T) \Leftrightarrow \forall i, j, k\colon \big(i \leqslant j \leqslant k \wedge H_{p(i)} = H_{p(k)} = \text{content}(T)\big) \Rightarrow p(i) = p(j);$$

$$\mathbf{Cons}(p, T) \Leftrightarrow \forall i\colon \text{content}\big(T[i]\big) \subseteq H_{p(i)};$$

$$\mathbf{SMon}(p,T) \Leftrightarrow \forall i,j\colon i<j \Rightarrow H_{p(i)} \subseteq H_{p(j)};$$

$$\mathbf{Mon}(p,T) \Leftrightarrow \forall i,j\colon i<j \Rightarrow \mathrm{content}(T) \cap H_{p(i)} \subseteq \mathrm{content}(T) \cap H_{p(j)};$$

$$\mathbf{WMon}(p,T) \Leftrightarrow \forall i,j\colon i<j \wedge \mathrm{content}\big(T[j]\big) \subseteq H_{p(i)} \Rightarrow H_{p(i)} \subseteq H_{p(j)};$$

$$\mathbf{Caut}(p,T) \Leftrightarrow \forall i,j\colon H_{p(i)} \subsetneq H_{p(j)} \Rightarrow i \leqslant j;$$

$$\mathbf{Caut}_{\mathbf{Tar}}(p,T) \Leftrightarrow \forall i\colon \neg\big(\mathrm{content}(T) \subsetneq H_{p(i)}\big);$$

$$\mathbf{Wb}(p,T) \Leftrightarrow \forall i,j\colon \big(\exists k\colon i<k\leqslant j \wedge p(i)\neq p(k)\big)$$
$$\Rightarrow \big(\mathrm{content}\big(T[j]\big) \cap H_{p(j)}\big) \setminus H_{p(i)} \neq \emptyset;$$

$$\mathbf{Conv}(p,T) \Leftrightarrow \forall i,j\colon \big(i\leqslant j \wedge \mathrm{content}\big(T[j]\big) \subseteq H_{p(i)}\big) \Rightarrow p(i)=p(j);$$

$$\mathbf{SemConv}(p,T) \Leftrightarrow \forall i,j\colon \big(i\leqslant j \wedge \mathrm{content}\big(T[j]\big) \subseteq H_{p(i)}\big) \Rightarrow H_{p(i)}=H_{p(j)};$$

$$\mathbf{Dec}(p,T) \Leftrightarrow \forall i,j,k\colon (i\leqslant j\leqslant k \wedge H_{p(i)}=H_{p(k)}) \Rightarrow H_{p(i)}=H_{p(j)};$$

$$\mathbf{SDec}(p,T) \Leftrightarrow \forall i,j,k\colon (i\leqslant j\leqslant k \wedge H_{p(i)}=H_{p(k)}) \Rightarrow p(i)=p(j).$$

We combine any two learning restrictions δ and δ' by intersecting them, which is denoted by their juxtaposition. With **T** we define the learning restriction which is always true and interpret it as absence of a learning restriction.

Finally, a *learning criterion* is a tuple $(\alpha, \mathcal{C}, \beta, \delta)$, where α and δ are learning restrictions, $\mathcal{C}$ is the set of admissible learners, usually $\mathcal{P}$ or $\mathcal{R}$, and β is an interaction operator. We write $\tau(\alpha)\mathcal{C}\mathbf{Txt}\beta\delta$ to denote the learning criterion and omit $\mathcal{C}$ if it equals $\mathcal{P}$, and a learning restriction if it equals **T**. Let $h \in \mathcal{C}$ be an admissible learner. We say that h $\tau(\alpha)\mathcal{C}\mathbf{Txt}\beta\delta$-learns a language L with respect to some hypothesis space $\mathcal{H}$ if and only if, for all texts $T \in \mathbf{Txt}$, we have $\alpha(\beta(h,T),T)$ and, for all $T \in \mathbf{Txt}(L)$, $\delta(\beta(h,T),T)$. Note that α has to hold on *all* texts, while δ only has to be true on texts of the learnable language L. The set of languages $\tau(\alpha)\mathcal{C}\mathbf{Txt}\beta\delta$-learned by h with respect to some hypothesis space $\mathcal{H}$ is denoted by $\tau(\alpha)\mathcal{C}\mathbf{Txt}\beta\delta(h)$. The class of such sets over all admissible learners is denoted by $[\tau(\alpha)\mathcal{C}\mathbf{Txt}\beta\delta]_{\mathbf{ind}}$, the so-called *learning power* of $\tau(\alpha)\mathcal{C}\mathbf{Txt}\beta\delta$-learners.

2.2. Normal forms

To prove certain statements on a learner, there are properties that come in handy. For example, except for **Cons**, all considered learning restrictions are *delayable*. Intuitively, a learning restriction is delayable if it allows for arbitrary, but finite postponing of hypotheses [21]. Formally, a learning restriction is delayable if and only if for all sequences of hypotheses p, texts $T, T' \in \mathbf{Txt}$ with $\mathrm{content}(T) = \mathrm{content}(T')$ and non-decreasing, unbounded functions $r\colon \mathbb{N} \to \mathbb{N}$, if we have $\delta(p,T)$ and, for all $n \in \mathbb{N}$, $\mathrm{content}(T[r(n)]) \subseteq \mathrm{content}(T'[n])$, then also $\delta(p \circ r, T')$ holds.

A common property of the considered learning restrictions is that they solely depend on the semantic of the hypotheses and on the position of mind changes. This property is formalized in the notion of *pseudo-semantic* learning restrictions [23]. A learning restriction δ is pseudo-semantic if and only if for all learning sequences p and texts $T \in \mathbf{T}$, if $\delta(p,T)$ and for a learning sequence p', with, for all $n \in \mathbb{N}$, $p(n)$ and $p'(n)$ are defined and semantically equivalent and $p(n) = p(n+1)$ implies $p'(n) = p'(n+1)$, then $\delta(p',T)$. All considered learning restrictions are pseudo-semantic.

We regularly make use of *locking sequences*. These are sequences that contain enough information such that a given learner, after seeing this sequence, suggests a correct hypothesis for the target language and does not change its mind whatever data from the target language it is given. Formally, let h be a **G**-learner and $\mathcal{H} = (H_i)_{i\in\mathbb{N}}$ a hypothesis space. Then a sequence $\sigma \in \mathbb{S}\mathrm{eq}$ is a locking sequence for h on a language L, if, for all sequences $\tau \in L^*_\#$, we have $h(\sigma) = h(\sigma^\frown\tau)$ and $H_{h(\sigma^\frown\tau)} = L$ [5]. For **Bc**-learners, we drop the first requirement and call σ a **Bc**-*locking sequence* [16]. This definition can directly be expanded to learners with other interaction operators. Let h be such a learner and consider its starred learner h^*. Then, a sequence σ is called a locking sequence for h on L if and only if σ is a locking sequence for h^* on L. We remark that, in the case of partially set-driven and set-driven learners, we refer to locking sequences as *locking information* and *locking set*, respectively. Note that, if a learner learns a language there always exists a (**Bc**-) locking sequence [5], but there exist texts where no initial sequence thereof is a (**Bc**-) locking sequence. Given a learner h and a language L it learns, if on any text $T \in \mathbf{Txt}(L)$ there exists an initial sequence thereof which is a (**Bc**-) locking sequence for h on L, we call h *strongly (***Bc***-) locking on* L. If h is strongly (**Bc**-) locking on every language it learns, we call h *strongly (***Bc***-) locking* [21].

3. Learning indexed families without hypothesis spaces

In this section, we present a useful result on which we build our remaining results. When learning indexed families with respect to (arbitrary) hypothesis spaces, the choice of the latter is crucial for successful learning. However, it is often a non-trivial task to construct the fitting hypothesis space. With Theorem 1, we show that one can forgo this necessity and, so to speak, obtain the hypothesis space on the run.

We make use of so-called C*-indices* or *characteristic indices* [4]. Intuitively, a C-index of a language L is a program for its characteristic function. Formally, an index e is a C-index of the language L if and only if $\varphi_e \equiv \chi_L$. We also denote $C_e = \{x \in \mathbb{N} | \varphi_e(x) = 1.\}$. Note that if e is a C-index of L then $C_e = L$. Now, we can request a learner to converge to a C-index instead of an index with respect to some hypothesis space. Exemplary, when requiring syntactic convergence to a C-index we write, for all sequences of hypotheses p and all texts T,

$$\mathbf{Ex}_C(p, T) \Leftrightarrow \exists n_0 \forall n \geqslant n_0\colon\ p(n) = p(n_0) \wedge C_{p(n_0)} = \mathrm{content}(T).$$

Adjusting the other considered restrictions to cover C-indices is immediate and, thus, omitted. For clarity, we write $\tau(\alpha)\mathcal{C}\mathbf{Txt}\beta\delta_C$ in case of learning C-indices. Analogously, we denote with $[\tau(\alpha)\mathcal{C}\mathbf{Txt}\beta\delta_C]$ the set of all classes $\tau(\alpha)\mathcal{C}\mathbf{Txt}\beta\delta_C$-learnable by some learner h.

In particular, we show that learners which output characteristic indices on any input may be translated into total learners which learn with respect to a hypothesis space. To that end, we define the restriction **CInd** [4], where the learner must output C-indices. Formally, for any hypothesis sequence p and any text T, we have

$$\mathbf{CInd}(p, T) \Leftrightarrow \forall i, x\colon \varphi_{p(i)}(x)\downarrow \wedge \varphi_{p(i)}(x) \in \{0, 1\}.$$

We show the equality of the two learning approaches. Firstly, the output when learning with respect to a hypothesis space can easily be interpreted as a characteristic index. For the other direction, we consider all hypotheses output by the $\tau(\mathbf{CInd})$-learner, that is, the learner which outputs C-indices on any input, as hypothesis space and choose the right (minimal) index of this hypothesis space to maintain successful learning.

Theorem 1. *Let α, δ be pseudo-semantic restrictions, let $\beta \preceq \mathbf{G}$ be an interaction operator and let $\mathcal{L}$ be an indexed family. Then, $\mathcal{L}$ is $\tau(\mathbf{CInd}\alpha)\mathbf{Txt}\beta\delta_C$-learnable by some learner h if and only if there exist a total learner h' and a hypothesis space $\mathcal{H}$ such that $h'\tau(\alpha)\mathbf{Txt}\beta\delta$-learns $\mathcal{L}$ with respect to $\mathcal{H}$.*

Proof. For the first direction, let $\mathcal{L} \in [\tau(\mathbf{CInd}\alpha)\mathbf{Txt}\beta\delta_C]$ be an indexed family learned by a $\tau(\mathbf{CInd}\alpha)\mathbf{Txt}\beta\delta_C$-learner h. Let h^* be the starred form of h, that is, the **G**-learner simulating h. As h is $\tau(\mathbf{CInd})$, so is h^* and we can define the indexed hypothesis space $\mathcal{H} = (C_{h^*(\sigma)})_{\sigma \in \mathbb{S}\mathrm{eq}}$. As h learns $\mathcal{L}$, we have $\mathcal{L} \subseteq \mathcal{H}$. Fix an order $\leqslant$ on the set of all finite sequences. Then, we define the learner h', for notational convenience in its starred form, for any finite sequence σ as

$$\left(h'\right)^*(\sigma) = \min_{\leqslant}\left\{\sigma' \in \mathbb{S}\mathrm{eq} | h^*\left(\sigma'\right) = h^*(\sigma)\right\}.$$

Note that the min-search terminates as σ is a candidate thereof. Intuitively, $(h')^*(\sigma)$ returns the $\leqslant$-least sequence σ' such that h^* does not change, that is, $h^*(\sigma) = h^*(\sigma')$. This sequence is then an index for the hypothesis space $\mathcal{H}$. Now, for two sequences σ and τ, we have $h^*(\sigma) = h^*(\tau)$ if and only if $(h')^*(\sigma) = (h')^*(\tau)$. Also, $h^*(\sigma)$ and $(h')^*(\sigma)$ are semantically equivalent. Thus, $h'\tau(\alpha)\mathbf{Txt}\beta\delta$-learns $\mathcal{L}$ with respect to $\mathcal{H}$.

Conversely, let $\mathcal{L}$ be such that there exist a total learner h' and an indexed hypothesis space $\mathcal{H} = (L_j)_{j\in\mathbb{N}}$ such that $h'\tau(\alpha)\mathbf{Txt}\beta\delta$-learns $\mathcal{L}$ with respect to $\mathcal{H}$. We provide a learner h which $\tau(\mathbf{CInd}\alpha)\mathbf{Txt}\beta\delta_C$-learns $\mathcal{L}$. Let $(h')^*$ and h^* denote their starred forms. As $\mathcal{H}$ is an indexed hypothesis space, there exists a total computable function f such that for all $j, x \in \mathbb{N}$

$$f(j, x) = \begin{cases} 1, & x \in L_j; \\ 0, & \text{otherwise.} \end{cases}$$

Due to the S-m-n Theorem, there exists a strictly monotonically increasing function $g \in \mathcal{R}$ such that, for all $j, x \in \mathbb{N}$, we have $\varphi_{g(j)}(x) = f(j, x)$. Now, we define, for all finite sequences $\sigma \in \mathbb{S}\mathrm{eq}$,

$$h^*(\sigma) = g\big((h')^*(\sigma)\big).$$

We conclude the proof by showing that h $\tau(\mathbf{CInd}\alpha)\mathbf{Txt}\beta\delta_C$-learns $\mathcal{L}$. We first show that h is $\tau(\mathbf{CInd})$. This follows immediately as, for any finite sequence σ, there exists $j \in \mathbb{N}$ such that

$$\varphi_{h^*(\sigma)}(x) = \varphi_{g(j)}(x) = f(j, x) = \begin{cases} 1, & x \in L_j; \\ 0, & \text{otherwise.} \end{cases}$$

As h' only makes mind changes when h does and as, for any $\sigma \in \mathbb{S}\mathrm{eq}$, $L_{(h')^*(\sigma)} = C_{h^*(\sigma)}$, we have that h $\tau(\alpha)\mathbf{Txt}\beta\delta_C$-learns $\mathcal{L}$. □

We will see that in many cases requiring the learner h' to be total is not restrictive. It is already known that, when learning arbitrary classes of recursively enumerable languages, Gold-style learners, obeying delayable learning restrictions, may be assumed total [21]. This result directly transfers to learning indexed families with respect to some hypothesis space.

Theorem 2 ([21]). *For delayable restrictions δ we get $[\mathcal{R}\mathbf{TxtG}\delta]_{\mathbf{ind}} = [\mathbf{TxtG}\delta]_{\mathbf{ind}}$.*

Proof. This proof follows [21] and is included for completeness. Immediately, we get the inclusion $[\mathcal{R}\mathbf{TxtG}\delta]_{\mathbf{ind}} \subseteq [\mathbf{TxtG}\delta]_{\mathbf{ind}}$. For the other, let h $\mathbf{TxtG}\delta$-learn $\mathcal{L}$ with respect to a hypothesis space $\mathcal{H}$. Let $e \in \mathbb{N}$ such that $h = \varphi_e$. To define the equivalent learner, let Φ be a Blum complexity measure [6], that is for example, for $e, x \in \mathbb{N}$, $\Phi_e(x)$ could be the number of steps the program e needs to halt on input x. We define a learner h' such that, for all sequences $\sigma \in \mathbb{S}\mathrm{eq}$,

$$h'(\sigma) = h\Big(\max_{\subseteq}\big(\big\{\sigma' \subseteq \sigma \mid \Phi_e\big(\sigma'\big) \leqslant |\sigma|\big\} \cup \{\varepsilon\}\big)\Big).$$

As we only allow total learning sequences of h for languages in $\mathcal{L}$, we have $h(\varepsilon)\downarrow$ and, thus, h' is indeed total and computable. We show that h $\mathcal{R}\mathbf{TxtG}\delta$-learns $\mathcal{L}$ with respect to $\mathcal{H}$. To that end, we use that δ is delayable. Let $L \in \mathcal{L}$ and $T \in \mathbf{Txt}(L)$. Now, for all $n \in \mathbb{N}$, let $r(n) = |\max_{\subseteq}(\{\sigma' \subseteq T[n] | \Phi_e(\sigma') \leqslant n.\} \cup \{\varepsilon\})|$. Note that, for all $n \in \mathbb{N}$, we have $h'(T[n]) = h(T[r(n)])$. As δ is delayable, it suffices to show that r is non-decreasing and unbounded to prove that $h'\mathcal{R}\mathbf{TxtG}\delta$-learns $\mathcal{L}$ with respect to $\mathcal{H}$. By definition of r, we have that r is non-decreasing and, for all $n \in \mathbb{N}$, we have $r(n) \leqslant n$ and that r is unbounded, as there exists $m \in \mathbb{N}$ with $m \geqslant n$ such that $\Phi_e(T[n]) \leqslant m$ and, thus $r(m) \geqslant n$. This concludes the proof. □

4. Consistent learning

Certain learners exhibit various properties which ease mathematical proofs and provide insight in the necessary learning behavior. One such is being consistent with the information given while maintaining learning power. For example, various behaviorally correct learners have been investigated for consistency [23]. We study whether this can also be assumed when learning indexed families and also with explanatory learners. Initial results can be found within the literature [1,27–29]. We complete these using condense results (see Lemmas 4 to 6) and obtain Theorem 3, the main result of this section. For the remainder this section, we fix $\Gamma := \{\mathbf{G}, \mathbf{Psd}, \mathbf{Sd}\}$ and $\Delta := \{\mathbf{Ex}, \mathbf{Bc}\}$.

Theorem 3. *For all* $\delta \in \{\mathbf{T}, \mathbf{Mon}, \mathbf{SMon}, \mathbf{WMon}, \mathbf{Caut_{Tar}}, \mathbf{SemConv}, \mathbf{Conv}, \mathbf{NU}, \mathbf{SNU}, \mathbf{Dec}, \mathbf{SDec}\}$*, all* $\delta' \in \Delta$ *and all* $\beta \in \Gamma$*, we have*

$$[\tau(\mathbf{Cons})\mathbf{Txt}\beta\delta\delta']_{\mathbf{ind}} = [\mathcal{R}\mathbf{Txt}\beta\delta\delta']_{\mathbf{ind}}.$$

Unrestricted **Bc**-learners can be made consistent by simply patching the missing elements into the hypothesis. As one can check for consistency, one can decide whether changing the hypothesis is necessary or not. It is immediate to see that this strategy works out as, firstly, the padding needs only to be done while the learner did not converge yet and, secondly, as the learner needs not to serve any additional requirements. Note that this also preserves **Ex**-convergence. Interestingly, the same idea also works out for strongly monotone and monotone learners. Especially here, Theorem 1 comes in handy as we create the hypothesis space containing the padded hypotheses on the way.

Lemma 4. *For* $\beta \in \Gamma$*,* $\delta \in \{\mathbf{T}, \mathbf{Mon}, \mathbf{SMon}\}$ *and* $\delta' \in \Delta$*, we have*

$$[\tau(\mathbf{Cons})\mathbf{Txt}\beta\delta\delta']_{\mathbf{ind}} = [\mathcal{R}\mathbf{Txt}\beta\delta\delta']_{\mathbf{ind}}.$$

Proof. The inclusion $[\tau(\mathbf{Cons})\mathbf{Txt}\beta\delta\delta']_{\mathbf{ind}} \subseteq [\mathcal{R}\mathbf{Txt}\beta\delta\delta']_{\mathbf{ind}}$ is immediate. For the other, we use a construction which patches in the seen data while maintaining the given learning restriction, as seen when learning of arbitrary classes [23]. By Theorem 1, it suffices to show

$$[\tau(\mathbf{CInd})\mathbf{Txt}\beta\delta\delta'_C] \subseteq [\tau(\mathbf{CIndCons})\mathbf{Txt}\beta\delta\delta'_C].$$

Let h be a learner and let $\mathcal{L} \subseteq \tau(\mathbf{CInd})\mathbf{Txt}\beta\delta\delta'_C(h)$. We define a $\tau(\mathbf{CIndCons})\mathbf{Txt}\beta\delta\delta'_C$-learner h' using some auxiliary functions. For ease of notation, we use h and h' as their starred learners. Due to the S-m-n Theorem, there exists $s \in \mathcal{R}$ such that, for all $x \in \mathbb{N}$ and all finite sequences σ,

$$\varphi_{s(\sigma)}(x) = \begin{cases} 1, & \text{if } x \in \text{content}(\sigma) \vee \varphi_{h(\sigma)}(x) = 1; \\ 0, & \text{otherwise.} \end{cases}$$

We now define the learner h' such that for any finite sequence σ

$$h'(\sigma) = \begin{cases} h(\sigma), & \text{if content}(\sigma) \subseteq C_{h(\sigma)}; \\ s(\sigma), & \text{otherwise.} \end{cases}$$

Note that $\varphi_{s(\sigma)}$ and h' are total because h is a $\tau(\mathbf{CInd})$-learner. Intuitively, h' has the same hypothesis as h, if this hypothesis is consistent. Otherwise, it patches the input set into the hypothesis of h. Thus, by construction, h' only outputs consistent C-indices, i.e., it is a $\tau(\mathbf{CIndCons})$-learner. In particular, note that for any sequence σ we have that

$$C_{h'(\sigma)} = C_{h(\sigma)} \cup \text{content}(\sigma). \tag{1}$$

It remains to be shown that $h'\delta'$-learns every language in $\mathcal{L}$ and that it obeys the restriction δ while doing so. We first show δ'-convergence. Let $L \in \mathcal{L}$ and $T \in \mathbf{Txt}(L)$. As h learns L, there exists $n_0 \in \mathbb{N}$ such that, for all $n \geqslant n_0$, we have $C_{h(T[n])} = L$ and, in the case of $\delta' = \mathbf{Ex}$, also $h(T[n]) = h(T[n_0])$. For $n \geqslant n_0$, as $h(T[n])$ is consistent, $h'(T[n])$ will output $h(T[n])$, proving that $h'\delta'$-learns L from text T.

Lastly, we prove that h' learns $\mathcal{L}$ without violating the restriction δ. For $\delta = \mathbf{T}$ this follows immediately. We consider the remaining restrictions separately. Let $L \in \mathcal{L}$ and let $T \in \mathbf{Txt}(L)$.

1. Case: $\delta = \mathbf{SMon}$. Let $n, m \in \mathbb{N}$ such that $n \leqslant m$. Since h is **SMon**, we have that

$$C_{h(T[n])} \subseteq C_{h(T[m])}.$$

Now, by Equation (1), we get that h' is **SMon** as

$$C_{h'(T[n])} = C_{h(T[n])} \cup \operatorname{content}\bigl(T[n]\bigr) \subseteq C_{h(T[m])} \cup \operatorname{content}\bigl(T[m]\bigr) = C_{h'(T[m])}.$$

2. Case: $\delta = \mathbf{Mon}$. Let $n, m \in \mathbb{N}$ such that $n \leqslant m$. Since h is **Mon**, we have that

$$C_{h(T[n])} \cap \operatorname{content}(T) \subseteq C_{h(T[m])} \cap \operatorname{content}(T).$$

Now, by Equation (1), we get that h' is **Mon** as

$$\begin{aligned} C_{h'(T[n])} \cap \operatorname{content}(T) &= \bigl(C_{h(T[n])} \cup \operatorname{content}\bigl(T[n]\bigr)\bigr) \cap \operatorname{content}(T) \\ &\subseteq \bigl(C_{h(T[m])} \cup \operatorname{content}\bigl(T[m]\bigr)\bigr) \cap \operatorname{content}(T) \\ &= C_{h'(T[m])} \cap \operatorname{content}(T). \end{aligned}$$

This concludes the proof. □

The former strategy does not work for target-cautious learners. The reason is that by simply adding missing elements, one can suddenly overgeneralize the target language. “Resetting” the conjecture to solely the information given when determining non-consistency preserves target-cautiousness, as we show in the next result. Interestingly, this strategy of “resetting” also works for weakly monotone learners as they, when inconsistent, may propose new suggestions.

Lemma 5. *For $\beta \in \Gamma$, $\delta \in \{\mathbf{WMon}, \mathbf{Caut_{Tar}}\}$ and $\delta' \in \Delta$, we have*

$$\bigl[\tau(\mathbf{Cons})\mathbf{Txt}\beta\delta\delta'\bigr]_{\mathbf{ind}} = \bigl[\mathcal{R}\mathbf{Txt}\beta\delta\delta'\bigr]_{\mathbf{ind}}.$$

Proof. The inclusion $[\tau(\mathbf{Cons})\mathbf{Txt}\beta\delta\delta']_{\mathbf{ind}} \subseteq [\mathcal{R}\mathbf{Txt}\beta\delta\delta']_{\mathbf{ind}}$ is immediate. For the other, it suffices to show $[\tau(\mathbf{CInd})\mathbf{Txt}\beta\delta\delta'_C] \subseteq [\tau(\mathbf{CIndCons})\mathbf{Txt}\beta\delta\delta'_C]$, due to Theorem 1. We use a similar construction as used for the case of W-indices [23]. The idea is to output solely the content of the given data if the original learner is not consistent. Let h be a learner and let $\mathcal{L} \subseteq \tau(\mathbf{CInd})\mathbf{Txt}\beta\delta\delta'_C(h)$. We define a learner h' which $\tau(\mathbf{CIndCons})\mathbf{Txt}\beta\delta\delta'_C$-learns $\mathcal{L}$. For ease of notation, we use h and h' as their starred learners. We now define the learner h', such that for any finite sequence $\sigma \in \mathbb{S}\mathrm{eq}$

$$h'(\sigma) = \begin{cases} h(\sigma), & \text{if } \operatorname{content}(\sigma) \subseteq C_{h(\sigma)}; \\ \operatorname{ind}(\operatorname{content}(\sigma)), & \text{otherwise.} \end{cases}$$

Note that h' is total and computable because h only outputs C-indices. By construction, h' only outputs consistent C-indices, i.e., it is a $\tau(\mathbf{CIndCons})$-learner.

It remains to be shown that $h'\delta'$-learns every language in $\mathcal{L}$ while obeying the restriction δ. We first show δ'-convergence. Let $L \in \mathcal{L}$ and $T \in \mathbf{Txt}(L)$. As h learns L, there exists n_0 such that, for all $n \geqslant n_0$, we have $C_{h(T[n])} = L$ and, in the case of $\delta' = \mathbf{Ex}$, also $h(T[n]) = h(T[n_0])$. For $n \geqslant n_0$, as $h(T[n])$ is consistent, $h'(T[n])$ will output $h(T[n])$, proving that $h'\delta'$-learns L on text T.

Lastly, we prove that h' satisfies the restriction δ. We consider the restrictions separately. Let $L \in \mathcal{L}$ and let $T \in \mathbf{Txt}(L)$.

1. Case: $\delta = \mathbf{WMon}$. Let $n, m \in \mathbb{N}$ such that $n \leqslant m$ and $\mathrm{content}(T[m]) \subseteq C_{h'(T[n])}$. We show that $C_{h'(T[n])} \subseteq C_{h'(T[m])}$. If $h(T[n])$ is not consistent, that is, $\mathrm{content}(T[n]) \nsubseteq C_{h(T[n])}$, then $C_{h'(T[n])} = \mathrm{content}(T[n])$. Thus, we have that

$$C_{h'(T[n])} = \mathrm{content}\big(T[n]\big) \subseteq \mathrm{content}\big(T[m]\big) \subseteq C_{h'(T[m])}.$$

Otherwise, $h(T[n])$ is consistent and, thus, $C_{h(T[n])} = C_{h'(T[n])}$. Since, by assumption, $\mathrm{content}(T[m]) \subseteq C_{h'(T[n])}$ and since h is weakly monotone, we have that $C_{h(T[n])} \subseteq C_{h(T[m])}$ and also that $C_{h(T[m])}$ is consistent. Thus, in this case we get

$$C_{h'(T[n])} = C_{h(T[n])} \subseteq C_{h(T[m])} = C_{h'(T[m])}.$$

2. Case: $\delta = \mathbf{Caut_{Tar}}$. Let $n \in \mathbb{N}$, then $h'(T[n])$ outputs either $h(T[n])$, in which case the hypothesis is target-cautious due by assumption, or it outputs the conjecture $\mathrm{ind}(\mathrm{content}(T[n]))$. As $\mathrm{content}(T[n]) \subseteq \mathrm{content}(T)$, this hypothesis is also target-cautious. □

Although (semantically) conservative learners may change their mind when being inconsistent with the data given, the above strategy does not work. The problem is that one may make them consistent too early and, thus, prevent later mind changes from happening. An interesting strategy solves the problem. One mimics the (possibly) inconsistent learner on information without repetition. Learning is preserved this way, as when inferring infinite target languages there will always be new information to correct an incorrect conjecture. On the other hand, learning finite target languages is also successful since, given all information without repetition, either the learner was correct anyway or making it consistent is a correct guess.

Lemma 6. *For $\beta \in \Gamma$, $\delta \in \{\mathbf{SemConv}, \mathbf{Conv}\}$ and $\delta' \in \Delta$, we have*

$$\big[\tau(\mathbf{Cons})\mathbf{Txt}\beta\delta\delta'\big]_{\mathbf{ind}} = \big[\mathcal{R}\mathbf{Txt}\beta\delta\delta'\big]_{\mathbf{ind}}.$$

Proof. The inclusion $[\tau(\mathbf{Cons})\mathbf{Txt}\beta\delta\delta']_{\mathbf{ind}} \subseteq [\mathcal{R}\mathbf{Txt}\beta\delta\delta']_{\mathbf{ind}}$ is immediate. For the other, we use a similar construction as when learning W-indices [23]. By Theorem 1, it suffices to show $[\tau(\mathbf{CInd})\mathbf{Txt}\beta\delta\delta'_C] \subseteq [\tau(\mathbf{CIndCons})\mathbf{Txt}\beta\delta\delta'_C]$. Let h be a learner and let $\mathcal{L} \subseteq \tau(\mathbf{CInd})\mathbf{Txt}\beta\delta\delta'_C(h)$. We define a learner h' which $\tau(\mathbf{CIndCons})\mathbf{Txt}\beta\delta\delta'_C$-learns $\mathcal{L}$. For ease of notation, we use h and h' as their starred learners. Given a sequence σ, we write $\tilde{\sigma}$ for the sequence without repetitions or pause symbols. Analogously, **Psd**-learners given $(\mathrm{content}(\sigma), |\sigma|)$ consider $(\mathrm{content}(\tilde{\sigma}), |\tilde{\sigma}|)$ instead. Notably, **Sd**-learners receive the same information since we have $\mathrm{content}(\sigma) = \mathrm{content}(\tilde{\sigma})$. Now, we define h' such that, for all finite sequences $\sigma \in \mathbb{S}\mathrm{eq}$,

$$h'(\sigma) = \begin{cases} h(\tilde{\sigma}), & \text{if } \mathrm{content}(\tilde{\sigma}) \subseteq C_{h(\tilde{\sigma})}; \\ \mathrm{ind}(\mathrm{content}(\tilde{\sigma})), & \text{otherwise.} \end{cases}$$

Note that, by construction, h' is a $\tau(\mathbf{CIndCons})$-learner. The intuition for the learner h' is then to mimic h on information without repetition. This is important to ensure (semantic) conservativeness. Given σ, it either outputs the same hypothesis as $h(\tilde{\sigma})$, if this is consistent, or it outputs solely a C-index for the content of the input.

Next, we show that $h'\delta'$-learns $\mathcal{L}$. Let $L \in \mathcal{L}$ and $T \in \mathbf{Txt}(L)$. We distinguish whether L is finite or not.

1. Case: L is finite. Then, there exists a minimal $n_0 \in \mathbb{N}$ such that content$(T[n_0]) = L$. Then, by definition, for all $n \geqslant n_0$, we have that $h'(T[n_0]) = h'(T[n])$ as no new element will be witnessed. Now, if $h(T[n_0])$ is consistent, then, because h is (semantically) conservative and thus target-cautious, we have $C_{h(T[n_0])} = L$. Otherwise, h' outputs ind(content($T[n_0]$)). In both cases, $h'(T[n_0])$ is a correct hypothesis.
2. Case: L is infinite. Note that the transition to text T from the corresponding text $T' \in \mathbf{Txt}(L)$ which does not contain any duplicates or pause-symbols can be done using an unbounded, non-decreasing function $r\colon \mathbb{N} \to \mathbb{N}$, that is, $T = T' \circ r$. As δ' is delayable, it suffices to show the convergence on text T'. As h also converges on T', there exists some n_0 such that, for all $n \geqslant n_0$, we have $C_{h(T'[n])} = L$ and, if $\delta' = \mathbf{Ex}$, also $h(T'[n_0]) = h(T'[n])$. In particular, for all $n \geqslant n_0$, $h(T'[n])$ is consistent and, thus, $h'(T'[n]) = h(\widetilde{T'[n]}) = h(T'[n])$. Thus, $h'\delta'$-learns L from text T' since h does and, as δ' is delayable, h' also learns L from text T.

It remains to be shown that h' obeys δ. The basic idea is that h' may only make a mind change if it sees a new element which is not consistent with the current hypothesis. Formally, let $L \in \mathcal{L}$ and $T \in \mathbf{Txt}(L)$. Furthermore, let $n, m \in \mathbb{N}$, with $n < m$, be such that content$(T[m]) \subseteq C_{h'(T[n])}$. We distinguish between the two cases for δ.

1. Case: $\delta = \mathbf{SemConv}$. We show that $C_{h'(T[n])} = C_{h'(T[m])}$. In the case of content$(T[n]) =$ content$(T[m])$, this follows by definition. Otherwise, there exists an element in content$(T[m])$ which is not in content$(T[n])$. Thus, in order for $C_{h'(T[n])}$ to enumerate content$(T[m])$, that is, content$(T[m]) \subseteq C_{h'(T[n])}$, it must hold that content$(T[m]) \subseteq C_{h(\widetilde{T[n]})}$. Then, since h is semantically conservative, we have $C_{h(\widetilde{T[n]})} = C_{h(\widetilde{T[m]})}$. In particular, $h(\widetilde{T[m]})$ is consistent, meaning that $C_{h'(T[m])} = C_{h(\widetilde{T[m]})}$. Altogether, we get

$$C_{h'(T[n])} = C_{h(\widetilde{T[n]})} = C_{h(\widetilde{T[m]})} = C_{h'(T[m])}.$$

2. Case: $\delta = \mathbf{Conv}$. This case follows an analogous proof-idea, the main difference being that semantic equalities need to be replaced with syntactic ones. We show that $h'(T[n]) = h'(T[m])$. In the case of content$(T[n]) =$ content$(T[m])$, this follows by definition. Otherwise, there exists an element in content$(T[m])$ which is not in content$(T[n])$. Thus, in order for $C_{h'(T[n])}$ to enumerate content$(T[m])$, that is, content$(T[m]) \subseteq C_{h'(T[n])}$, it must hold that content$(T[m]) \subseteq C_{h(\widetilde{T[n]})}$. Then, since h is (syntactically) conservative, we have $h(\widetilde{T[n]}) = h(\widetilde{T[m]})$. In particular, $h(\widetilde{T[m]})$ is consistent, meaning that $h'(T[m]) = h(\widetilde{T[m]})$. Altogether, we get

$$h'\bigl(T[n]\bigr) = h\bigl(\widetilde{T[n]}\bigr) = h\bigl(\widetilde{T[m]}\bigr) = h'\bigl(T[m]\bigr),$$

which concludes the proof. □

We note that none of the above strategies work for learners which may not return to previous guesses, say, non-U-shaped learners, as one may, by patching missing elements into the hypotheses or "resetting" the hypotheses, accidentally produce a hypothesis for the target language and later discard it temporarily. However, another widely used strategy, called *poisoning* [8], works in this case. The main idea is to "poison" wrong hypotheses, that is, create obviously incorrect hypotheses. Since inconsistent hypotheses cannot be correct, we can poison those and obtain a consistent learner.

Lemma 7. *For $\beta \in \Gamma$, $\delta \in \{\mathbf{NU}, \mathbf{SNU}, \mathbf{Dec}, \mathbf{SDec}\}$ and $\delta' \in \Delta$, we have*

$$\bigl[\tau(\mathbf{Cons})\mathbf{Txt}\beta\delta\delta'\bigr]_{\mathbf{ind}} = \bigl[\mathcal{R}\mathbf{Txt}\beta\delta\delta'\bigr]_{\mathbf{ind}}.$$

Proof. We would like to express our gratitude to the anonymous reviewer who provided helpful feedback in developing the following proof. Observe that the inclusion $[\tau(\mathbf{Cons})\mathbf{Txt}\beta\delta\delta']_{\mathbf{ind}} \subseteq [\mathcal{R}\mathbf{Txt}\beta\delta\delta']_{\mathbf{ind}}$ is immediate. For the other direction, by Theorem 1, it suffices to show

$$\left[\tau(\mathbf{CInd})\mathbf{Txt}\beta\delta\delta'_C\right] \subseteq \left[\tau(\mathbf{CIndCons})\mathbf{Txt}\beta\delta\delta'_C\right].$$

Let h be a learner and let $\mathcal{L} \subseteq \tau(\mathbf{CInd})\mathbf{Txt}\beta\delta\delta'_C(h)$. We consider h in its starred form to prove the statement for all $\beta \in \Gamma$ at once. Consider the class of hypotheses made by h, that is, $\mathcal{H} = (C_{h^*(\sigma)})_{\sigma\in\mathbb{S}\mathrm{eq}}$. Intuitively, we construct an indexed family $\mathcal{F}$ containing poisoned hypotheses as follows. We start with a computable set F which is co-infinite, that is, $\mathbb{N} \setminus F$ is infinite, and which differs from all possible hypotheses made by h, that is, differs from all sets in $\mathcal{H}$, at infinitely many positions. We then extend this set with some finite information which grows as the data input grows. In particular, we will make it consistent. We then use these sets in case the original hypothesis was inconsistent (and therefore wrong).

For the formal proof, we use the following auxiliary notation. Given the interaction operator β and two finite sequences $\sigma, \tau \in \mathbb{S}\mathrm{eq}$, we write $\sigma \preceq_\beta \tau$ if and only if,

- in the case of $\beta = \mathbf{G}$, we have $\sigma \subseteq \tau$;
- in the case of $\beta = \mathbf{Psd}$, there exists a text $T \in \mathbf{Txt}$ and $n, m \in \mathbb{N}$, with $n \leqslant m$, such that $\mathrm{content}(T[n]) = \mathrm{content}(\sigma)$ and $\mathrm{content}(T[m]) = \mathrm{content}(\tau)$;
- in the case of $\beta = \mathbf{Sd}$, we have $\mathrm{content}(\sigma) \subseteq \mathrm{content}(\tau)$.

Intuitively, $\sigma \preceq_\beta \tau$ means that σ contains less information than τ according to β.

We construct $\mathcal{F}$ formally. Using a diagonalisation argument, we obtain a computable set F which is co-infinite, that is, $\mathbb{N} \setminus F$ is infinite, and which differs from, for all $\tau \in \mathbb{S}\mathrm{eq}$, $C_{h^*(\tau)}$ at infinitely many positions. Let m be a bijective computable function such that, for all $\sigma, \tau \in \mathbb{S}\mathrm{eq}$, $|\max(\mathrm{content}(\sigma))| \leqslant m(\sigma)$ and, if $\sigma \preceq_\beta \tau$ then $m(\beta(\sigma)) \leqslant m(\beta(\tau))$. Respecting the notation of starred learners, we write $m^*(\sigma) = m(\beta(\sigma))$. Intuitively, m creates a numbering which ensures that later information obtains a larger number.

For each $\sigma \in \mathbb{S}\mathrm{eq}$, let $\tilde{f} \in \mathcal{R}$ be such that $C_{\tilde{f}^*(\sigma)}$ is the set containing the first $m^*(\sigma)$ many elements which are not present in F. As F is co-infinite, we will always find such elements. Furthermore note that by the choice of m^* we have $\mathrm{content}(\sigma) \subseteq C_{\tilde{f}^*(\sigma)}$. Now, let $f \in \mathcal{R}$ be such that

$$C_{f^*(\sigma)} = F \cup C_{\tilde{f}^*(\sigma)}.$$

Note that $\mathrm{content}(\sigma) \subseteq C_{f^*(\sigma)}$. In the end, define $\mathcal{F} = (C_{f^*(\sigma)})_{\sigma\in\mathbb{S}\mathrm{eq}}$.

Now, given a finite sequence $\sigma \in \mathbb{S}\mathrm{eq}$ as input, consider the new learner g defined as

$$g^*(\sigma) = \begin{cases} h^*(\sigma), & \text{if } \mathrm{content}(\sigma) \subseteq C_{h^*(\sigma)}, \\ f^*(\sigma), & \text{otherwise.} \end{cases}$$

The learner g is a β-learner and consistent by definition. Furthermore, the learner g maintains δ'-learning as whenever h outputs a correct (and therefore consistent) hypothesis, g outputs the (syntactically) same hypothesis. We show that it obeys the restriction δ as well. Let $\sigma, \tau \in \mathbb{S}\mathrm{eq}$ be such that $\sigma \subseteq \tau$ and $C_{g^*(\sigma)} = C_{g^*(\tau)}$, which equals L in case δ equals **NU** and **SNU**. We show that for each $\rho \in \mathbb{S}\mathrm{eq}$ with $\sigma \subseteq \rho \subseteq \tau$, we have $C_{g^*(\sigma)} = C_{g^*(\rho)} = C_{g^*(\tau)}$ (in case δ equals **NU** and **Dec**) and $g^*(\sigma) = g^*(\rho) = g^*(\tau)$ (in case δ equals **SNU** and **SDec**).

If $\beta(\sigma) = \beta(\tau)$, we get the result immediately. Otherwise, assume $\beta(\sigma) \neq \beta(\tau)$. Thus, $C_{f^*(\sigma)} \neq C_{f^*(\tau)}$ and, by definition of g, the only possibility where $C_{g^*(\sigma)} = C_{g^*(\tau)}$ may happen is that $g^*(\sigma) = h^*(\sigma)$ and $g^*(\tau) = h^*(\tau)$. As h is a δ-learner, we have $C_{h^*(\sigma)} = C_{h^*(\rho)} = C_{h^*(\tau)}$. This, in particular implies

$$\mathrm{content}(\sigma) \subseteq \mathrm{content}(\rho) \subseteq \mathrm{content}(\tau) \subseteq C_{h^*(\tau)} = C_{h^*(\rho)}.$$

Thus, $g^*(\rho) = h^*(\rho)$ and, therefore, $C_{g^*(\sigma)} = C_{g^*(\rho)} = C_{g^*(\tau)}$ (in case δ equals **NU** and **Dec**) and $g^*(\sigma) = g^*(\rho) = g^*(\tau)$ (in case δ equals **SNU** and **SDec**). This means that g maintains δ. □

Altogether, the lemmas above give the proof of Theorem 3.

5. Delayable map for learning indexed families

In this section we compare the power of (possibly partial) learners underlying various delayable learning restrictions to each other. First, we gather known results from literature. It is a well-known fact that learners need time in order to obtain full learning power [27], that is, set-driven learners lack learning power. Furthermore, in the literature monotonic learners have been investigated thoroughly [27].

Theorem 8 ([27]). *We have* $[\mathbf{TxtGEx}]_{\mathbf{ind}} \setminus [\mathbf{TxtSdEx}]_{\mathbf{ind}} \neq \emptyset$ *and*

$$\begin{aligned}[\mathbf{TxtSdSMonEx}]_{\mathbf{ind}} &= [\mathbf{TxtGSMonEx}]_{\mathbf{ind}} \subsetneq [\mathbf{TxtSdMonEx}]_{\mathbf{ind}} \\ &\subsetneq [\mathbf{TxtGMonEx}]_{\mathbf{ind}} \subsetneq [\mathbf{TxtSdWMonEx}]_{\mathbf{ind}} \\ &= [\mathbf{TxtSdEx}]_{\mathbf{ind}} = [\mathbf{TxtGWMonEx}]_{\mathbf{ind}}.\end{aligned}$$

We remark that weak monotonicity as well as conservativeness is no restriction to set-driven learners [27]. We expand this result by showing that set-driven learners may be even assumed to be witness-based. This way, we also capture the remaining restrictions, such as (target-) cautiousness and (strong) decisiveness. To obtain the desired result, we first show that target-cautious and witness-based Gold-style learners possess the same learning power. The idea is that, as target-cautious learners never overgeneralize the target language, there always remain elements as witnesses to justify a mind change if the current hypothesis is wrong.

Theorem 9. *We have* $[\mathbf{TxtGWbEx}]_{\mathbf{ind}} = [\mathbf{TxtGCaut_{Tar}Ex}]_{\mathbf{ind}}$.

Proof. The inclusion $[\mathbf{TxtGWbEx}]_{\mathbf{ind}} \subseteq [\mathbf{TxtGCaut_{Tar}Ex}]_{\mathbf{ind}}$ is straightforward. For the other, by assuming that **G**-learners are total, see Theorem 2, and by Theorem 1, it suffices to show

$$\left[\tau(\mathbf{CInd})\mathbf{TxtCaut_{Tar}GEx}_C\right] \subseteq \left[\tau(\mathbf{CInd})\mathbf{TxtGWbEx}_C\right].$$

Let h be a learner with $\mathcal{L} \subseteq \tau(\mathbf{CInd})\mathbf{TxtGCaut_{Tar}Ex}_C(h)$. Using Theorem 3, we can assume h to be consistent, i.e., $\mathcal{L} \subseteq \tau(\mathbf{CIndCons})\mathbf{TxtCaut_{Tar}GEx}_C(h)$. We now prove that the following learner h' is a $\tau(\mathbf{CInd})\mathbf{TxtGWbEx}_C$-learner for $\mathcal{L}$. Let $h'(\varepsilon) = h(\varepsilon)$ and, for all finite $\sigma \in \mathbb{S}\mathrm{eq}$ and $x \in \mathbb{N}$, let

$$h'(\sigma^\frown x) := \begin{cases} h'(\sigma), & \text{if } x \in C_{h'(\sigma)}; \\ h(\sigma^\frown x), & \text{otherwise.} \end{cases}$$

Intuitively, h' only updates its hypothesis if the latest datum may be used as a witness for a mind change. As h is also consistent, we immediately have that h' is witness-based. Furthermore, note that h' outputs a C-index on every input and thus is a $\tau(\mathbf{CInd})$-learner.

It remains to be shown that h' learns $\mathcal{L}$. To that end, let $L \in \mathcal{L}$ and $T \in \mathbf{Txt}(L)$. Since h correctly learns L there exists $n_0 \in \mathbb{N}$ such that, for all $n \geqslant n_0$, we have $h(T[n]) = h(T[n_0])$ and $\varphi_{h(T[n])} = \chi_L$. We distinguish the following cases.

1. Case: $h'(T[n_0])$ is a C-index for L. In this case, $L \setminus C_{h'(T[n_0])} = \emptyset$, and thus h' cannot change its mind again. Thus, it converges correctly.
2. Case: $h'(T[n_0])$ is no C-index for L. As h is target-cautious and h' mimics h, it cannot hold $L \subsetneq C_{h'(T[n_0])}$. Thus, there exists $x \in L$ with $x \notin C_{h'(T[n_0])}$. By definition of h' and by consistency of h, we have $x \notin \mathrm{content}(T[n_0])$. Let n_1 be such that $x \in \mathrm{content}(T[n_1])$. Then, by construction, for all $n \geqslant n_1$, we have that $h'(T[n]) = h(T[n_1])$, which is a C-index for L. □

This equality also includes weakly monotone learners. Thus, we already have that these are as powerful as set-driven learners. However, we go one step further and show that these learners may even be assumed total. We make use of Theorems 1 and 2. The idea is to mimic the Gold-style learner on the minimal, consistent hypothesis. This way, target-cautiousness is preserved as well as the learning power, since no guess overgeneralizes the target language and, thus, checking for consistency is a valid strategy.

Theorem 10. *We have* $[\mathcal{R}\mathbf{TxtSdCaut_{Tar}Ex}]_{\mathbf{ind}} = [\mathbf{TxtGCaut_{Tar}Ex}]_{\mathbf{ind}}$.

Proof. The inclusion $[\mathcal{R}\mathbf{TxtSdCaut_{Tar}Ex}]_{\mathbf{ind}} \subseteq [\mathbf{TxtGCaut_{Tar}Ex}]_{\mathbf{ind}}$ is immediate. For the other one, as **G**-learner may be assumed total, see Theorem 2, and by Theorem 1, it suffices to show

$$\big[\tau(\mathbf{CInd})\mathbf{TxtGCaut_{Tar}Ex}_C\big] \subseteq \big[\tau(\mathbf{CInd})\mathbf{TxtSdCaut_{Tar}Ex}_C\big].$$

Let h be a learner and let $\mathcal{L} \subseteq \tau(\mathbf{CInd})\mathbf{TxtGCaut_{Tar}Ex}_C(h)$. By Theorem 3, we may assume h to be consistent. For a finite set D, for $k \leqslant |D|$, let $\sigma_D[k]$ be the canonical sequence of D of length k, that is, the sequence of the first k elements in D in strictly ascending order, and define the **Sd**-learner h' as

$$h'(D) = h\big(\sigma_D\big[\min\big\{k' \leqslant |D| \,|\, D \subseteq C_{h(\sigma_D[k'])}\big\}\big]\big).$$

Mimicking learner h, the newly defined learner h' is target-cautious whenever h is and it always outputs C-indices. It remains to be shown that $h'\mathbf{SdEx}_C$-learns $\mathcal{L}$. Let, to that end, $L \in \mathcal{L}$. We distinguish the following cases.

1. Case: L is finite. Let $k_0 \in \mathbb{N}$ be the minimal $k' \leqslant |L|$ such that $L \subseteq C_{h(\sigma_L[k'])}$. By consistency of h, such k' exists. Then, we have by h being target-cautious that $\neg(L \subsetneq C_{h(\sigma_L[k_0])})$. Altogether, we have

$$C_{h'(L)} = C_{h(\sigma_L[k_0])} = L.$$

2. Case: L is infinite. Then, consider the canonical text T of L. As h learns L, there exists a minimal $n_0 \in \mathbb{N}$ such that $C_{h(T[n_0])} = L$. By target-cautiousness of h and minimal choice of n_0, there exists $n_1 \geqslant n_0$ such that for all $n < n_0$ we have

$$\operatorname{content}\big(T[n_1]\big) \setminus C_{h(T[n])} \neq \emptyset.$$

Then, for all D with $\operatorname{content}(T[n_1]) \subseteq D \subseteq L$, we have $h'(D) = h(T[n_0])$ as desired. □

Again with Theorem 1, we obtain that set-driven learners may be assumed total and witness-based. The idea resembles the approach for partially set-driven learners of arbitrary classes of languages [22]. To obtain this, we assume the information coming in a certain order and then mimic the learner on the least input where no mind change is witnessed. Then, while enumerating, we check whether any later datum causes a mind change. If so, we stop the enumeration. Especially here, Theorem 1 comes in handy as we do not need to fix the hypothesis space beforehand, but rather build it up on the fly.

Theorem 11. *We have* $[\mathcal{R}\mathbf{TxtSdWbEx}]_{\mathbf{ind}} = [\mathcal{R}\mathbf{TxtSdCaut_{Tar}Ex}]_{\mathbf{ind}}$.

Proof. The direction $[\mathcal{R}\mathbf{TxtSdWbEx}]_{\mathbf{ind}} \subseteq [\mathcal{R}\mathbf{TxtSdCaut_{Tar}Ex}]_{\mathbf{ind}}$ follows immediately. For the other, by Theorem 1, it suffices to show that

$$\big[\tau(\mathbf{CInd})\mathbf{TxtSdCaut_{Tar}Ex}_C\big] \subseteq \big[\tau(\mathbf{CInd})\mathbf{TxtSdWbEx}_C\big].$$

Let h $\tau(\mathbf{CInd})\mathbf{TxtSdCaut_{Tar}Ex}_C$-learn $\mathcal{L}$. We define the desired witness-based learner h'. Given a finite set D and $k \leqslant |D|$, let $D[k]$ be the set of the first k elements (in ascending order) in D, and define

$$k_D = \min\big\{k \leqslant |D| \,|\, \forall D', D[k] \subseteq D' \subseteq D \colon h\big(D'\big) = h(D)\big\}.$$

That is, $D[k_D]$ contains the minimal amount of elements of D in ascending order where no mind change is witnessed. In what follows, we use $C^x_{h(\sigma)}$ to denote all elements in $C_{h(\sigma)}$ up until x, that is, $C^x_{h(\sigma)} = \{x' \leqslant x | \varphi_{h(\sigma)}(x') = 1\}$. Then, for any finite set D, define

$$\varphi_{s(D)}(x) = \begin{cases} 1, & \text{if } x \in D; \\ 0, & \text{else, if } \varphi_{h(D)}(x) = 0; \\ 1, & \text{else, if } \forall D', D \subseteq D' \subseteq D \cup C^x_{h(D)}\colon h(D) = h'(D); \\ 0, & \text{otherwise.} \end{cases}$$

Note that for any locking set D of some language L, we have $C_{s(D)} = L$. Then, for any finite set D, we define the learner

$$h'(D) = \begin{cases} \mathrm{ind}(D[k_D]), & \text{if } \exists x < \max(D[k_D]), x \notin D[k_D]\colon \varphi_{h(D[k_D])}(x) = 1; \\ s(D[k_D]), & \text{otherwise.} \end{cases}$$

Intuitively, the learner first searches the minimal amount of elements where no mind change is witnessed. Then, given $D[k_D]$, if the learner on input $D[k_D]$ witnesses an element to be (possibly) out of order, it outputs $\mathrm{ind}(D[k_D])$. This way, we keep this element as a witness for a possible, later mind change. Otherwise, the learner outputs $s(D[k_D])$ which conducts a forward search and enumerates all elements where no mind change is witnessed.

Formally, we first show that h' learns $\mathcal{L}$ correctly. Let therefore $L \in \mathcal{L}$. We distinguish the following cases.

1. Case: L is finite. Here, $h(L)$ is a correct conjecture and, thus, $h(L[k_L])$ as well. Note that, in particular, $L[k_L]$ is a locking set for L. As there exists no $x < \max(L[k_L])$ with $\varphi_{h(L[L_k])}(x) = 1$, we have that $h'(L[k_L]) = s(L[k_L])$. Since $L[k_L]$ is a locking set, $s(L[k_L])$ is a correct conjecture.
2. Case: L is infinite. Let T_c be the canonical text for L and let $n_0 \in \mathbb{N}$ be minimal such that $D_0 := \mathrm{content}(T_c[n_0])$ is a locking set for h on L. As, for all $n < n_0$, $\mathrm{content}(T_c[n])$ is no locking set, there exists some $x_n \in L$ where this is witnessed. Let $x_{\max} = \max\{x_n | n < n_0\}$ and let $n_1 \geqslant n_0$ such that, for $D_{n_1} := \mathrm{content}(T_c[n_1])$, we have $x_{\max} \in D_{n_1}$. Then, for any D with $D_{n_1} \subseteq D \subseteq L$, we have that $h'(D) = h'(D_{n_1})$. As D_{n_1} is a locking set, we have that $h'(D_{n_1}) = s(D_{n_1}[k_{D_{n_1}}])$ is a correct hypothesis.

Lastly, we show that h' is witness-based. Let, to that end, $D_1 \subseteq D_2 \subseteq D_3 \subseteq L$ such that $h'(D_1) \neq h'(D_2)$. We show that $(C_{h'(D_3)} \cap D_3) \setminus C_{h'(D_1)} \neq \emptyset$. For $i \in \{1, 2, 3\}$, let $k_i := k_{D_i}$ and let $D'_i := D[k_i]$. Then, as $h'(D_i) = h'(D'_i)$, it suffices to show

$$(C_{h'(D'_3)} \cap D_3) \setminus C_{h'(D'_1)} \neq \emptyset.$$

We distinguish the following cases.

1. Case: $D'_1 = D'_3$. In particular, $D'_1 = D'_2 = D'_3$. Then, $h'(D_1) = h'(D_2)$, a contradiction to the initial assumption.
2. Case: $D'_3 \setminus D'_1 \neq \emptyset$. Let x be a maximal such element. Either, $x \notin C_{h(D'_1)}$ and it will not be considered when enumerating $C_{h'(D'_1)}$. Otherwise, $x \notin C_{h'(D'_1)}$ as it either is less than $\max(D'_1)$ or it will not be enumerated by $s(D'_1)$ as it witnesses a mind change.
3. Case: $D'_1 \setminus D'_3 \neq \emptyset$. If $D'_3 \subseteq D'_1$, then, as $D_1 \subseteq D_3$, the minimality of k_1 is violated. Thus, it also holds that $D'_3 \setminus D'_1 \neq \emptyset$, and we proceed just as in the previous case. □

This completes the study of set-driven learners following delayable learning restrictions. It remains to be shown that Gold-style learners may be assumed strongly decisive. We do so in two steps. First, we show that unrestricted learners may be assumed strongly non-U-shaped in general. The idea is to search for locking sequences. If we witness that the current sequence is not locking, we *poison* the produced hypothesis [8]. We can do so, as indexed families provide a decision procedure to check whether $x \in L_i$ or not. When poisoning, we simply output a hypothesis contradicting all of the given languages. Note that, by Theorem 1, we may construct poisoned hypotheses on the fly.

Theorem 12. *We have* $[\mathbf{TxtGSNUEx}]_{\mathbf{ind}} = [\mathbf{TxtGEx}]_{\mathbf{ind}}$.

Proof. The inclusion $[\mathbf{TxtGSNUEx}]_{\mathbf{ind}} \subseteq [\mathbf{TxtGEx}]_{\mathbf{ind}}$ is immediate. For the other direction, note that **G**-learners may be assumed total by Theorem 2. Thus, by Theorem 1, it suffices to show that

$$\left[\tau(\mathbf{CInd})\mathbf{TxtGEx}_C\right] \subseteq \left[\tau(\mathbf{CInd})\mathbf{TxtGSNUEx}_C\right].$$

Let h be a $\tau(\mathbf{CInd})\mathbf{TxtGEx}_C$-learner and let $\mathcal{L} \subseteq \tau(\mathbf{CInd})\mathbf{TxtGEx}_C(h)$. We provide a $\tau(\mathbf{CInd})\mathbf{TxtGSNUEx}_C$-learner for $\mathcal{L} = (L_i)_{i\in\mathbb{N}}$. The idea is the following. Since $\mathcal{L}$ is indexed, there exists a procedure to decide whether $x \in L_i$ or not. Given any input, we check whether it serves as a locking sequence. Note that **TxtGEx**-learners may be assumed strongly locking [14]. While it does so, we mimic the learner on this input. Once we figure it not being a locking sequence, we start *poisoning* this guess such that it does not refer to any of the possible languages L_i. Thus, the resulting learner will output the correct language once it finds a locking sequence thereof.

Formally, we first define the auxiliary predicate which, given a sequence σ and an element $x \in \mathbb{N}$, tells us whether σ is a candidate for a locking sequence up until the element x, that is,

$$Q(\sigma, x) = \begin{cases} 1, & \text{if } \exists \sigma' \in (C^x_{h(\sigma)})^{\leqslant x}_{\#}, \sigma \subseteq \sigma' \exists y \leqslant x : \varphi_{h(\sigma)}(y) \neq \varphi_{h(\sigma')}(y); \\ 0, & \text{otherwise.} \end{cases}$$

Again, we use $C^x_{h(\sigma)}$ to denote all elements in $C_{h(\sigma)}$ up until x, that is, $C^x_{h(\sigma)} = \{x' \leqslant x \mid \varphi_{h(\sigma)}(x') = 1\}$. Next, the S-m-n Theorem provides us with an auxiliary function which poisons conjectures on non-locking sequences. There exists $s \in \mathcal{R}$ such that for all $x \in \mathbb{N}$ and $\sigma \in \mathbb{S}\mathrm{eq}$

$$\varphi_{s(\sigma)}(x) = \begin{cases} \varphi_{h(\sigma)}(x), & \text{if } Q(\sigma, x) = 0; \\ 0, & \text{else, if } x \in L_{x - \min\{y \in \mathbb{N} \mid Q(\sigma, y) = 1\}}; \\ 1, & \text{otherwise.} \end{cases}$$

Note that in the second case $\{y \in \mathbb{N} \mid Q(\sigma, y) = 1\}$ is non-empty (as the first case does not hold) and that its elements are bounded by x. Lastly, we need the following auxiliary function which finds the minimal sequence on which h agrees with the current hypothesis up until some point. For any sequence σ, define

$$M(\sigma) = \left\{\sigma' \subseteq \sigma \mid \forall \sigma'' \in \mathrm{content}(\sigma)^{\leqslant|\sigma|}_{\#} \forall x \leqslant |\sigma| : \varphi_{h(\sigma)}(x) = \varphi_{h(\sigma'^\frown\sigma'')}(x)\right\}.$$

Finally, we define the learner h' as, for any sequence σ,

$$h'(\sigma) = s\big(\min\big(M(\sigma)\big)\big).$$

Now, let $L \in \mathcal{L}$ and let $T \in \mathbf{Txt}(L)$. We show that h' converges to a correct hypothesis and, afterwards, show the learner to be strongly non-U-shaped. As h is strongly locking, there exists a minimal $n_0 \in \mathbb{N}$ such that $T[n_0]$ is a locking sequence for h on L. In particular, there exists $n_1 \geqslant n_0$ such that, for all $n < n_0$, $T[n] \notin M(T[n_1])$, that is, all sequences prior to $T[n_0]$ are not locking. Then, for all $n \geqslant n_1$, we have that $\min(M(T[n])) = T[n_0]$ and, thus, $h'(T[n]) = s(T[n_0])$. Furthermore, for any $x \in \mathbb{N}$, we have that $Q(T[n_0], x) = 0$ as all the sequences output the same hypothesis. Thus, $\varphi_{s(T[n_0])} = \varphi_{h(T[n_0])}$, meaning that $s(T[n_0])$ is a C-index for L.

We now show that successful learning is also strongly non-U-shaped. First, we show, for all $n < n_0$, that $s(T[n])$ is no C-index for L. By minimality of n_0, $T[n]$ is no locking sequence for h on L. Now, if $h(T[n])$ is no C-index of

L, neither will $s(T[n])$ be, as it either outputs the same as $h(T[n])$ or eventually contradicts all languages in $\mathcal{L}$. If, otherwise, $h(T[n])$ is a C-index of L, there exists some point $x \in \mathbb{N}$ witnessing $T[n]$ not to be a locking sequence. Then, $s(T[n])$ starts contradicting all languages in $\mathcal{L}$. Thus, $s(T[n])$ and also $h'(T[n])$ is no C-index for L. □

Building on this result, we go one step further and show the learners to be even strongly decisive. The strategy the newly found learner employs is to wait before changing its hypothesis until it witnesses a mind change. And, when doing so, it first checks whether this mind change produces a new hypothesis which is different from all previous ones.

Theorem 13. *We have* $[\mathbf{TxtGSDecEx}]_{\mathbf{ind}} = [\mathbf{TxtGSNUEx}]_{\mathbf{ind}}$.

Proof. The inclusion $[\mathbf{TxtGSDecEx}]_{\mathbf{ind}} \subseteq [\mathbf{TxtGSNUEx}]_{\mathbf{ind}}$ follows immediately. For the other, it suffices, by the observation that **G**-learners may be assumed total (Theorem 2) and by Theorem 1, to show that

$$\left[\tau(\mathbf{CInd})\mathbf{TxtGSNUEx}_C\right] \subseteq \left[\tau(\mathbf{CInd})\mathbf{TxtGSDecEx}_C\right].$$

To that end, let h be a learner and let $\mathcal{L} \subseteq \tau(\mathbf{CInd})\mathbf{TxtGSNUEx}_C(h)$. We define an equivalently powerful $\tau(\mathbf{CInd})\mathbf{TxtGSDecEx}_C$-learner h' as follows. Let $h'(\varepsilon) = h(\varepsilon)$ and, for any finite sequence $\sigma \neq \varepsilon$, let $\sigma' \subsetneq \sigma$ be the minimal sequence on which $h'(\sigma') = h'(\sigma^-)$, that is, the sequence on which h' based its previous output. Then, define

$$h'(\sigma) = \begin{cases} h(\sigma'), & \text{if } \forall \sigma'', \sigma' \subseteq \sigma'' \subseteq \sigma : h(\sigma') = h(\sigma''); \\ h(\sigma), & \text{else, if } \forall \sigma'' \subseteq \sigma'\ \exists x \leqslant |\sigma| : \varphi_{h'(\sigma'')}(x) \neq \varphi_{h(\sigma)}(x); \\ h(\sigma'), & \text{otherwise.} \end{cases}$$

As h' mimics h, h' always outputs C-indices and, hence, is $\tau(\mathbf{CInd})$. The intuition is to only update the hypothesis if the current hypothesis cannot be based on a locking sequence and if all previous ones are witnessed to be semantically different. As h is **SNU**, h' may never abandon a correct guess and all hypotheses before that are incorrect. Thus, h' preserves the learning power.

Formally, we first show that h' is, indeed, **SDec**. We do so by showing that whenever h' makes a mind change, this new hypothesis is certainly semantically different from all previous ones and, thus, also syntactically different. Let $L \in \mathcal{L}$ and let $\sigma \in L_\#^*$ such that $h'(\sigma^-) \neq h'(\sigma)$, that is, h made a mind change. Note that $h'(\sigma) = h(\sigma)$. Furthermore, let h' base its prior hypothesis on $\sigma' \subseteq \sigma^-$, that is, $\sigma' \subseteq \sigma^-$ is the minimal sequence on which, for all σ'' with $\sigma' \subseteq \sigma'' \subseteq \sigma^-$, we have $h'(\sigma'') = h'(\sigma^-)$. The only case where h' makes a mind change is, if for all $\sigma'' \subseteq \sigma'$ there exists $x \leqslant |\sigma|$ such that

$$\varphi_{h'(\sigma'')}(x) \neq \varphi_{h(\sigma)}(x).$$

As $h'(\sigma) = h(\sigma)$ and, therefore, $\varphi_{h(\sigma)} = \varphi_{h'(\sigma)}$, we have, for all $\tilde{\sigma} \subseteq \sigma'$,

$$C_{h'(\tilde{\sigma})} \neq C_{h'(\sigma)}.$$

As there are no further mind changes until σ^-, this holds for all $\tilde{\sigma} \subsetneq \sigma$. Thus, h' is **SDec**.

To show that h converges correctly, let $L \in \mathcal{L}$ and let $T \in \mathbf{Txt}(L)$. Then there exists a (minimal) n_0 such that, for all $n \geqslant n_0$, $h(T[n_0]) = h(T[n])$ and $h(T[n])$ is a C-index for L. We distinguish the following cases.

1. Case: $h'(T[n_0]) = h(T[n_0])$. In this case, $h'(T[n_0])$ is a correct C-index and, as h never changes its mind again, neither does h'.

2. Case: $h'(T[n_0]) \neq h(T[n_0])$. Let $n_1 < n_0$ be such that $h'(T[n_0]) = h(T[n_1])$. In particular, $h(T[n_1]) \neq h(T[n_0])$. Thus, the first case of the definition of h' cannot hold. By h being **SNU** and by the minimal choice of n_0, there exists some minimal $n_2 \geqslant n_0$ such that h' witnesses all hypotheses prior to (and including) $h(T[n_1])$ to differ from $h(T[n_2])$. Then, by the second case of the definition, it will output $h(T[n_2])$ never to change its mind again. □

Altogether, we obtain the full map as depicted in Fig. 1(a). It remains open to include partially set-driven learners into this picture. Due to known results from literature and the results we obtain, it remains to be shown whether Gold-style learners may be assumed strongly decisive and partially set-driven at the same time. We pose the following open question.

Open Problem 1. May Gold-style strongly decisive learners be assumed partially set-driven so?

6. Comparing convergence criteria

In this section we compare *total* learners under various memory constraints which converge syntactically to such that converge semantically. The gathered results we depict in Fig. 1(b). Usually, semantically converging learners are more powerful than their syntactic counterpart. This can be observed, for example, when learning arbitrary classes of languages [14]. However, when learning indexed families class-comprisingly different results are obtained.

We first gather known results which can be easily obtained from learning with (possibly) partial learners, by either applying Theorem 2 or slightly adapting the known proof. In the latter case, we provide the proofs for completeness reasons. It is known that explanatory **G**-learners and behaviorally correct ones are equally powerful [29]. By Theorem 2, this also holds true for total learners (compare Equation (2)). Furthermore, it is known that Gold-style learners do not rely on the order of the presented elements but rather on the time given [1,14,33] (see Equation (3)). By patching in the information given [7,10,23], even iterative **Bc**-learners are as powerful as Gold-style **Bc**-learners (see Equation (4)). As patching changes the hypothesis with every new datum, this approach does not work for explanatory iterative learners. It is known that this problem cannot be solved as there exists a well-known class separating set-driven explanatory learners from iterative ones [16,26] (compare Equation (5)). On the other hand, each iterative learner can be made set-driven by simply, given all data, mimicking the iterative learner on input with pause-symbols between each two elements [18,26] (compare Equation (6)).

Theorem 14. *We have*

$$[\mathcal{R}\mathbf{TxtGEx}]_{\mathbf{ind}} = [\mathcal{R}\mathbf{TxtGBc}]_{\mathbf{ind}}, \quad [29], \tag{2}$$

$$[\mathcal{R}\mathbf{TxtPsdEx}]_{\mathbf{ind}} = [\mathbf{TxtGEx}]_{\mathbf{ind}}, \quad [1,14,33], \tag{3}$$

$$[\mathcal{R}\mathbf{TxtItBc}]_{\mathbf{ind}} = [\mathcal{R}\mathbf{TxtGBc}]_{\mathbf{ind}}, \quad [7,10,23], \tag{4}$$

$$[\mathcal{R}\mathbf{TxtSdEx}]_{\mathbf{ind}} \setminus [\mathcal{R}\mathbf{TxtItEx}]_{\mathbf{ind}} \neq \emptyset, \quad [16,26], \tag{5}$$

$$[\mathcal{R}\mathbf{TxtItEx}]_{\mathbf{ind}} \subseteq [\mathcal{R}\mathbf{TxtSdEx}]_{\mathbf{ind}}, \quad [18,26]. \tag{6}$$

Proof. Most of these results are either known or immediately obtainable. We include the proofs for completeness reasons.

Eq. (2): It is known that $[\mathbf{TxtGEx}]_{\mathbf{ind}} = [\mathbf{TxtGBc}]_{\mathbf{ind}}$ [29]. With Theorem 2, we get the desired.

Eq. (3): The inclusion $[\mathcal{R}\mathbf{TxtPsdEx}]_{\mathbf{ind}} \subseteq [\mathbf{TxtGEx}]_{\mathbf{ind}}$ is straightforward. For the other, we follow the proof of the analogous result when learning classes of recursively enumerable languages [14]. Let h **TxtGEx**-learn $\mathcal{L}$ with respect to a hypothesis space $\mathcal{H}$. Without losing generality, see Theorem 2, let h be total. We define a **Psd**-learner h' using an auxiliary function $M \in \mathcal{R}$ as, for all finite sets $D \subseteq \mathbb{N}$ and $t \in \mathbb{N}$,

$$M(D,t) = \left\{\sigma \in D_{\#}^{\leqslant t} \mid \forall \tau \in D_{\#}^{\leqslant t} : h(\sigma) = h(\sigma\tau)\right\};$$

$$h'(D,t) = \begin{cases} h(\min(M(D,t))), & \text{if } M(D,t) \neq \emptyset; \\ h(\varepsilon), & \text{otherwise.} \end{cases}$$

Intuitively, h' mimics h on minimal potential locking sequences. Note that h' is total as h is so. To show that h learns $\mathcal{L}$, let $L \in \mathcal{L}$ and $T \in \mathbf{Txt}(L)$. Let σ_0 be the minimal locking sequence of h on L. We show that h' eventually converges to $h(\sigma_0)$. To that end, let $n_0 \in \mathbb{N}$ be large enough such that, with $D_0 = \text{content}(T[n_0])$, we have

- $\text{content}(\sigma_0) \subseteq D_0$,
- $\sigma_0 \leqslant n_0$ and
- for all $\sigma < \sigma_0$ there exists $\sigma' \in (D_0)_\#^{\leqslant n_0}$ such that $h(\sigma) \neq h(\sigma\sigma')$, i.e., σ' witnesses $\sigma \notin M(D_0, n_0)$.

Then, for all $n \geqslant n_0$, we have $\min(M(\text{content}(T[n]), n)) = \sigma_0$ and thus h' converges to $h(\sigma_0)$. As this is a correct hypothesis for L, h' learns $\mathcal{L}$.

Eq. (4): Immediately, we have $[\mathcal{R}\mathbf{TxtItBc}]_{\mathbf{ind}} \subseteq [\mathcal{R}\mathbf{TxtGBc}]_{\mathbf{ind}}$. We apply a padding argument [23] for the other direction. By Theorem 1, it suffices to show that

$$[\tau(\mathbf{CInd})\mathbf{TxtGBc}_C] \subseteq [\tau(\mathbf{CInd})\mathbf{TxtItBc}_C].$$

Let h be a $\tau(\mathbf{CInd})\mathbf{TxtGBc}_C$-learner and let $\mathcal{L} \subseteq \tau(\mathbf{CInd})\mathbf{TxtGBc}_C(h)$. Recall that pad is an injective padding function such that for all $e \in \mathbb{N}$ and all finite sequences σ we have $\varphi_{\text{pad}(e,\sigma)} = \varphi_e$. We define the iterative learner h' for all previous hypotheses p, all finite sequences σ and all $x \in \mathbb{N}$,

$$h'(\emptyset) = \text{pad}(h(\varepsilon), \varepsilon);$$
$$h'(\text{pad}(p,\sigma), x) = h'(\text{pad}(h(\sigma^\frown x), \sigma^\frown x)).$$

It is immediate to see that, for all sequences σ, we have $\varphi_{(h')^*(\sigma)} = \varphi_{h(\sigma)}$. Thus, $h'\tau(\mathbf{CInd})\mathbf{TxtItBc}$-learns $\mathcal{L}$.

Eq. (5): This is a standard proof and we include it for completeness [16]. By Theorem 1, it suffices to provide a class of languages $\mathcal{L}$ which is $\tau(\mathbf{CInd})\mathbf{TxtSdEx}_C$-learnable but not $\tau(\mathbf{CInd})\mathbf{TxtItEx}_C$ so. We define $\mathcal{L} := \{\mathbb{N} \setminus \{0\}\} \cup \{D \cup \{0\} | D \subseteq_{\mathbf{Fin}} \mathbb{N}\}$. Then, the following learner learns $\mathcal{L}$. Fix p_0 as a code for the language $\mathbb{N} \setminus \{0\}$ and define, for any finite sequence σ,

$$h(\sigma) = \begin{cases} p_0, & \text{if } 0 \notin \text{content}(\sigma); \\ \text{ind}(\text{content}(\sigma)), & \text{otherwise.} \end{cases}$$

It is immediate that h learns $\mathcal{L}$. Now, assume there exists an iterative learner h' which $\tau(\mathbf{CInd})\mathbf{TxtItEx}$-learns $\mathcal{L}$. Let $L = \mathbb{N} \setminus \{0\}$, let T be a text of L and let $n_0 \in \mathbb{N}$ such that for all $n \geqslant n_0$ we have $h'(T[n_0]) = h'(T[n])$. Let $x = \max(\text{content}(T[n_0]))$, then on the following two texts of distinct languages in $\mathcal{L}$

$$T_1 = \sigma^\frown(x'+1)^\frown 0^\infty;$$
$$T_2 = \sigma^\frown(x'+2)^\frown 0^\infty,$$

the learner h' generates the same hypotheses. Thus, it is unable to distinguish between these two. Therefore, $\mathcal{L}$ cannot be learned by h'.

Eq. (6): This is a standard proof and we include it for completeness [18]. Let h **TxtItEx**-learn the indexed family $\mathcal{L}$ with respect to $\mathcal{H}$. We provide a **Sd**-learner learning $\mathcal{L}$. To that end, we expand the hypothesis space $\mathcal{H}$ by adding all finite sets. This new hypothesis space we denote by $\mathcal{H}'$. For ease of notation, we refer to these new indices as, for all D, $\text{ind}(D)$. Now, for any set D, let $\text{sort}_\#(D)$ be the sequence of the elements in D sorted in ascending order, with a # between each two elements. Furthermore, let h^* be the starred form of h. Now, we define h' as, for all finite sets D,

$$h'(D) = \begin{cases} h^*(\text{sort}_\#(D)), & \text{if } h^*(\text{sort}_\#(D)) = h^*(\text{sort}_\#(D)^\frown\#); \\ \text{ind}(D), & \text{otherwise.} \end{cases}$$

To show that h' learns $\mathcal{L}$ with respect to $\mathcal{H}'$, let $L \in \mathcal{L}$. If L is finite, then either $h^*(\text{sort}_\#(L)) = h^*(\text{sort}_\#(L)^\frown\#)$, in which case h converges to $h'(L) = h^*(\text{sort}_\#(L))$ on text $\text{sort}_\#(L)^\frown\#^\infty$. Otherwise, we have $h'(L) = \text{ind}(L)$. In both cases, h' learns L as $h'(L)$ is a correct hypothesis for L.
On the other hand, if L is infinite, then h converges to a correct hypothesis for L on the text $\text{sort}_\#(L)$. Let σ_0 be the initial sequence of $\text{sort}_\#(L)$ on which h has converged and let $D_0 = \text{content}(\sigma_0)$. Then, for all $x \in \mathbb{N} \setminus D_0$, we have $h^*(\sigma_0{}^\frown x) = h^*(\sigma_0) = h^*(\sigma_0{}^\frown\#)$ as h is iterative. Therefore, for all D' with $D_0 \subseteq D' \subseteq L$, we have $h^*(\text{sort}_\#(D')) = h^*(\text{sort}_\#(D')^\frown\#)$ and $h^*(\text{sort}_\#(D')) = h^*(\text{sort}_\#(D_0))$, which is a correct hypothesis for L. As $h'(D') = h(\text{sort}_\#(D'))$, we have the convergence of h' to a correct hypothesis for L and, thus, h' learns L. □

Interestingly, only iterative learners benefit from loosening the convergence criterion. We have already investigated the situation for Gold-style and partially set-driven learners. Now, we conclude this section by showing that, first, total set-driven learners and then also transductive ones do not benefit from this relaxation.

Considering set-driven learners, we first show that behaviorally correct such learners may be assumed target-cautious in general. We do so by conducting a forward search, checking the learners output on each possible future hypothesis. Should we detect inconsistencies, we know that the current information is not locking and, thus, we can stop the enumeration. This way, no overgeneralization will happen as, otherwise, locking sets must be included in the search.

Lemma 15. *We have* $[\tau(\mathbf{Cons})\mathbf{TxtSdCaut_{Tar}Bc}]_{\mathbf{ind}} = [\mathcal{R}\mathbf{TxtSdBc}]_{\mathbf{ind}}$.

Proof. The inclusion $[\tau(\mathbf{Cons})\mathbf{TxtSdCaut_{Tar}Bc}]_{\mathbf{ind}} \subseteq [\mathcal{R}\mathbf{TxtSdBc}]_{\mathbf{ind}}$ is straightforward. By Theorem 1, for the other direction it suffices to show

$$[\tau(\mathbf{CInd})\mathbf{TxtSdBc}_C] \subseteq [\tau(\mathbf{CIndCons})\mathbf{TxtSdCaut_{Tar}Bc}_C].$$

We apply a similar construction of forward searches as when learning arbitrary classes of languages [12]. Let h be a total learner with $\mathcal{L} = \tau(\mathbf{CInd})\mathbf{TxtSdBc}_C(h)$. According to Theorem 3, we may assume h to be consistent on any input. Now, define a $\tau(\mathbf{CIndCons})\mathbf{TxtSdCaut_{Tar}Bc}_C$-learner h' as follows. Let, for all $x \in \mathbb{N}$ and finite sets $D \subseteq \mathbb{N}$,

$$E(x, D) := D \cup \{x\} \cup \{x' \leqslant x \mid \varphi_{h'(D)}(x') = 1\};$$

$$\varphi_{h'(D)}(x) = \begin{cases} 1, & \text{if } x \in D; \\ 0, & \text{else, if } \varphi_{h(D)}(x) = 0; \\ 1, & \text{else, if } \forall D'', D \subseteq D'' \subseteq E(x, D)\colon E(x, D) \subseteq C_{h(D'')}; \\ 0, & \text{otherwise.} \end{cases}$$

Intuitively, the conjecture of $h'(D)$ contains D itself and certain additional elements of the hypothesis of h on D. For these additional elements, h' checks whether all possible future hypotheses of h contain these elements as well.

If so, h' adds them in its hypothesis, otherwise it does not. This way, we prevent overgeneralizing target languages. Note that by construction h' is $\tau(\mathbf{CIndCons})$. Furthermore, note that, for any finite set D, we have

$$C_{h'(D)} \subseteq C_{h(D)}. \tag{7}$$

We first show that $h'\mathbf{Bc}$-learns any language $L \in \mathcal{L}$. We distinguish the following cases.

Case 1: L is finite. Since h learns L, we have $\varphi_{h(L)} = \chi_L$. Consider $h'(L)$. Now, for any element $x \in L$, we have $\varphi_{h'(L)}(x) = 1$, by definition. For any element $x \notin L$, we have $\varphi_{h(L)}(x) = 0$ and, therefore, $\varphi_{h'(L)}(x) = 0$ as well. Thus, $\varphi_{h'(L)} = \chi_L$ and h' learns L.

Case 2: L is infinite. Let D_0 be a $\mathbf{Bc}_C$-locking set for h on L. We show that, for any D with $D_0 \subseteq D \subseteq L$, $h'(D)$ is a correct hypothesis for L. We need to show that $L = C_{h'(D)}$. Note that, by Condition (7), $C_{h'(D)} \subseteq C_{h(D)} = L$. Thus, it remains to be shown that $L \subseteq C_{h'(D)}$. To that end, let $x \in L$. If $x \in D$, then $x \in C_{h'(D)}$ by consistency. Otherwise, we have $\varphi_{h(D)}(x) = 1$ and, thus, are in the third case of the definition of h'. We show that x gets enumerated this way. By Condition (7), we get

$$\begin{aligned} E(x, D) &= D \cup \{x\} \cup \{x' \leqslant x \mid \varphi_{h'(D)}(x') = 1\} \\ &\subseteq D \cup \{x\} \cup \{x' \leqslant x \mid \varphi_{h(D)}(x') = 1\} \subseteq L. \end{aligned}$$

As D_0, and therefore also D, is a $\mathbf{Bc}_C$-locking set, we have for all D'' with $D \subseteq D'' \subseteq L$ that

$$E(x, D) \subseteq L = C_{h(D'')}.$$

So the third condition is met and, therefore, $\varphi_{h'(D)}(x) = 1$. Hence, $h'\mathbf{Bc}_C$-learns L.

Finally, it remains to be shown that h' is target-cautious. To that end, assume that h' is not target-cautious. Thus, there exists a language $L \in \mathcal{L}$ and a set $D \subseteq L$ such that $L \subsetneq C_{h'(D)}$. Let $\tilde{x} \in C_{h'(D)} \setminus L$ and let $D_0 \supseteq D$ be a $\mathbf{Bc}_C$-locking set for L on h. Let $x' := \max((D_0 \cup \{\tilde{x}\}) \setminus D$. As $x' \in C_{h'(D)}$ but not in D, it must be enumerated by the third condition of the definition of h'. Note that $D_0 \subseteq E(x', D)$ (as D_0 must be enumerated until x'). Now, for all D'' with $D \subseteq D'' \subseteq E(x', D)$, it must hold that

$$\tilde{x} \in E(x', D_0) \subseteq C_{h(D'')}.$$

However, this is a contradiction for $D'' = D_0$ as $C_{h(D_0)} = L$ but $\tilde{x} \notin L$. This concludes the proof. □

In a second step, we construct an explanatory learner from the target-cautious behaviorally correct learner. The idea is to always mimic the $\mathbf{Bc}$-learner on the $\leqslant$-minimal set on which it is consistent. This way, we obtain syntactic convergence. On the other hand, the final hypothesis cannot be incorrect as, eventually, the learner has enough information to figure out incorrect guesses and, as it is target-cautious, these consistent conjectures are no overgeneralizations.

Theorem 16. *We have* $[\mathcal{R}\mathbf{TxtSdEx}]_{\mathbf{ind}} = [\mathcal{R}\mathbf{TxtSdBc}]_{\mathbf{ind}}$.

Proof. The inclusion $[\mathcal{R}\mathbf{TxtSdEx}]_{\mathbf{ind}} \subseteq [\mathcal{R}\mathbf{TxtSdBc}]_{\mathbf{ind}}$ is immediate. For the other, by Theorem 1, it suffices to show that

$$[\tau(\mathbf{CInd})\mathbf{TxtSdBc}_C] \subseteq [\tau(\mathbf{CInd})\mathbf{TxtSdEx}_C].$$

Let h be a learner and let $\mathcal{L} \subseteq \tau(\mathbf{CInd})\mathbf{TxtSdBc}_C(h)$. By Theorem 15, we may assume h to be target-cautious. We provide a learner h' which $\mathbf{Ex}_C$-learns $\mathcal{L}$. The main idea is to search for the first set on which the learner h is consistent. By its target-cautiousness, this way we will, eventually, conjecture the right language. For the formal details, fix a total order $\leqslant$ on finite sets. For any finite set D, we define the following auxiliary functions as

$$M(D) = \{D' \subseteq D \mid \forall x \in D\colon \varphi_{h(D')}(x) = 1\}.$$

Finally, for any finite set D, define $h'(D) = h(\min_{\leqslant}(M(D)))$. To show correctness, let $L \in \mathcal{L}$. We distinguish the following cases.

1. Case: L is finite. Let $D' \subseteq L$ such that $h'(L) = h(D')$. Since $D' \in M(L)$, we have $L \subseteq C_{h(D')}$. Since h is target-cautious, the equality holds, that is, $L = C_{h(D')}$. Thus, $h'(L) = h(D')$ is a correct hypothesis.
2. Case: L is infinite. Let $D_0 \subseteq L$ be the $\leqslant$-minimal set such that $C_{h(D_0)} = L$. Let $D_1 \supseteq D_0$ such that $D_1 \subseteq L$ and $\min_{\leqslant}(M(D_1)) = D_0$. Such a set exists as h, due to the minimal choice of D_0, conjectures incorrect guesses on $D' \subseteq L$ with $D' < D_0$ which do not overgeneralize the target language. Then, for any D, with $D_1 \subseteq D \subseteq L$, we have $h'(D) = h(D_0)$, a correct conjecture. □

Finally, behaviorally correct transductive learners, being unable to save any information about previous guesses, can be made explanatory immediately. One simply awaits a non-? guess and then checks for the first element in this guess which also produces a non-?.

Theorem 17. *We have* $[\mathcal{R}\mathbf{TxtTdEx}]_{\mathbf{ind}} = [\mathcal{R}\mathbf{TxtTdBc}]_{\mathbf{ind}}$.

Proof. The inclusion $[\mathcal{R}\mathbf{TxtTdEx}]_{\mathbf{ind}} \subseteq [\mathcal{R}\mathbf{TxtTdBc}]_{\mathbf{ind}}$ follows immediately. For the other, it suffices, by Theorem 1, to show that

$$[\tau(\mathbf{CInd})\mathbf{TxtTdBc}_C]_{\mathbf{ind}} \subseteq [\tau(\mathbf{CInd})\mathbf{TxtTdEx}_C]_{\mathbf{ind}}.$$

Let h be a $\tau(\mathbf{CInd})\mathbf{TxtTdBc}_C$-learner and let $\mathcal{L} \subseteq \tau(\mathbf{CInd})\mathbf{TxtTdBc}_C(h)$. We define $M \in \mathcal{R}$ and $h' \in \mathcal{R}$ such that for all $x \in \mathbb{N}$, $y \in \mathbb{N}_{\#}$,

$$M(x) = \{x' \leqslant x \mid \varphi_{h(x)}(x') = 1 \wedge h(x') \neq ?\};$$

$$h'(y) = \begin{cases} h(\#), & \text{if } y = \#; \\ ?, & \text{else, if } h(y) = ?; \\ h(\min(M(y))), & \text{otherwise.} \end{cases}$$

By construction, h' is only outputs C-indices (or ?). Intuitively, h' outputs the hypothesis of h on the smallest element in the hypothesis h on the current datum (if it is not "?"). We claim that h' learns $\mathcal{L}$. Let $L \in \mathcal{L}$. First, note that for any $x \in L$, we have that either $h(x) = ?$ or $h(x)$ is a C-index of L. If that were not the case, h would not identify L on any text which has infinitely many occurrences of x. Furthermore, for at least one $x \in L$, $h(x)$ must not be "?". Thus, there exists a minimal $x' \in L$, such that $h(x')$ is a characteristic index of L. The idea of this construction is to search for such minimal x'. Note that, if $h(x) \neq ?$, $M(x) \neq \emptyset$ as $x \in M(x)$.

Let $T \in \mathbf{Txt}(L)$ and let $n_0 \in \mathbb{N}$ be minimal such that $\mathrm{content}(T[n_0]) \neq \emptyset$ and such that $h(T(n_0 - 1)) \neq ?$. Then, $h'(T(n_0 - 1)) = h(\min(M(T(n_0 - 1))))$, a correct guess. Furthermore, for $n > n_0$ either $h(T(n)) = ?$ and with it $h'(T(n)) = ?$ or, otherwise, $h'(T(n)) = h(\min(M(T(n)))) = h(\min(M(T(n_0 - 1))))$. Thus, we have $\mathbf{Ex}_C$-convergence. □

We remark that all these results, except for Lemma 15 and Theorem 16, also hold for (possibly) partial learners. These are obtained either directly, for example by applying Theorem 2, or by slightly changing the provided proofs. However, one cannot directly translate Lemma 15, and therefore Theorem 16, as, in the forward search, totality of the learner is key. Otherwise, this search can be indefinite, breaking the indexability. We conclude this work by posing the following open question.

Open Problem 2. Does $[\mathbf{TxtSdEx}]_{\mathbf{ind}} = [\mathbf{TxtSdBc}]_{\mathbf{ind}}$ hold?

Acknowledgements

This work was supported by DFG Grant Number KO 4635/1-1. We would also like to thank the anonymous reviewers for their helpful feedback which helped improving this work.

References

[1] D. Angluin, Inductive inference of formal languages from positive data, *Information and Control* **45** (1980), 117–135. doi:10.1016/S0019-9958(80)90285-5.
[2] G. Baliga, J. Case, W. Merkle, F. Stephan and R. Wiehagen, When unlearning helps, *Information and Computation* **206** (2008), 694–709. doi:10.1016/j.ic.2007.10.005.
[3] J.M. Bārzdiņš, Inductive inference of automata, functions and programs, in: *American Mathematical Society Translations*, 1977, pp. 107–122.
[4] J. Berger, M. Böther, V. Doskoč, J. Gadea Harder, N. Klodt, T. Kötzing, W. Lötzsch, J. Peters, L. Schiller, L. Seifert, A. Wells and S. Wietheger, Learning languages with decidable hypotheses, in: *Computability in Europe (CiE)*, 2021, pp. 12813:25–12813:37.
[5] L. Blum and M. Blum, Toward a mathematical theory of inductive inference, *Information and Control* **28** (1975), 125–155. doi:10.1016/S0019-9958(75)90261-2.
[6] M. Blum, A machine-independent theory of the complexity of recursive functions, *Journal of the ACM* **14** (1967), 322–336. doi:10.1145/321386.321395.
[7] L. Carlucci, J. Case, S. Jain and F. Stephan, Results on memory-limited U-shaped learning, *Information and Computation* **205** (2007), 1551–1573. doi:10.1016/j.ic.2007.04.001.
[8] J. Case and T. Kötzing, Strongly non-U-shaped language learning results by general techniques, *Information and Computation* **251** (2016), 1–15. doi:10.1016/j.ic.2016.06.015.
[9] J. Case and C. Lynes, Machine inductive inference and language identification, in: *International Colloquium on Automata, Languages and Programming (ICALP)*, 1982, pp. 107–115. doi:10.1007/BFb0012761.
[10] J. Case and S.E. Moelius, U-shaped, iterative, and iterative-with-counter learning, *Machine Learning* **72**(1–2) (2008), 63–88.
[11] J. Case and S.E. Moelius, Optimal language learning from positive data, *Information and Computation* **209** (2011), 1293–1311. doi:10.1016/j.ic.2011.07.003.
[12] V. Doskoč and T. Kötzing, Cautious limit learning, in: *Algorithmic Learning Theory (ALT)*, 2020, pp. 117:251–117:276.
[13] M. Fulk, A Study of Inductive Inference Machines, PhD thesis, 1985.
[14] M.A. Fulk, Prudence and other conditions on formal language learning, *Information and Computation* **85** (1990), 1–11. doi:10.1016/0890-5401(90)90042-G.
[15] E.M. Gold, Language identification in the limit, *Information and Control* **10** (1967), 447–474. doi:10.1016/S0019-9958(67)91165-5.
[16] S. Jain, D. Osherson, J.S. Royer and A. Sharma, *Systems That Learn: An Introduction to Learning Theory*, 2nd edn, MIT Press, Cambridge (MA), 1999.
[17] K. Jantke, Monotonic and non-monotonic inductive inference, *New Generation Computing* **8** (1991), 349–360. doi:10.1007/BF03037092.
[18] E.B. Kinber and F. Stephan, Language learning from texts: Mindchanges, limited memory, and monotonicity, *Information and Computation* **123** (1995), 224–241. doi:10.1006/inco.1995.1170.
[19] T. Kötzing, Abstraction and Complexity in Computational Learning in the Limit, PhD thesis, University of Delaware, 2009.
[20] T. Kötzing, A solution to Wiehagen's thesis, *Theory of Computing Systems* **60** (2017), 498–520. doi:10.1007/s00224-016-9678-0.
[21] T. Kötzing and R. Palenta, A map of update constraints in inductive inference, *Theoretical Computer Science* **650** (2016), 4–24. doi:10.1016/j.tcs.2016.07.028.
[22] T. Kötzing and M. Schirneck, Towards an atlas of computational learning theory, in: *Symposium on Theoretical Aspects of Computer Science (STACS)*, 2016, pp. 47:1–47:13.

[23] T. Kötzing, M. Schirneck and K. Seidel, Normal forms in semantic language identification, in: *Algorithmic Learning Theory (ALT)*, 2017, pp. 76:493–76:516.

[24] S. Lange and T. Zeugmann, Language learning in dependence on the space of hypotheses, in: *Conference on Computational Learning Theory (COLT)*, 1993, pp. 127–136.

[25] S. Lange and T. Zeugmann, Monotonic versus non-monotonic language learning, in: *Nonmonotonic and Inductive Logic*, 1993, pp. 254–269. doi:10.1007/BFb0030397.

[26] S. Lange and T. Zeugmann, Incremental learning from positive data, *Journal of Computer and System Sciences* **53** (1996), 88–103. doi:10.1006/jcss.1996.0051.

[27] S. Lange and T. Zeugmann, Set-driven and rearrangement-independent learning of recursive languages, *Mathematical Systems Theory* **29** (1996), 599–634. doi:10.1007/BF01301967.

[28] S. Lange, T. Zeugmann and S. Kapur, Monotonic and dual monotonic language learning, *Theoretical Computer Science* **155** (1996), 365–410. doi:10.1016/0304-3975(95)00284-7.

[29] S. Lange, T. Zeugmann and S. Zilles, Learning indexed families of recursive languages from positive data: A survey, *Theoretical Computer Science* **397** (2008), 194–232. doi:10.1016/j.tcs.2008.02.030.

[30] D.N. Osherson, M. Stob and S. Weinstein, Learning strategies, *Information and Control* **53** (1982), 32–51. doi:10.1016/S0019-9958(82)91097-X.

[31] D.N. Osherson and S. Weinstein, Criteria of language learning, *Information and Control* **52** (1982), 123–138. doi:10.1016/S0019-9958(82)80025-9.

[32] H. Rogers Jr., *Theory of Recursive Functions and Effective Computability*, MIT Press, Cambridge (MA), 1987.

[33] G. Schäfer-Richter, Über Eingabeabhängigkeit und Komplexität von Inferenzstrategien, PhD thesis, RWTH Aachen University, Germany, 1984.

[34] K. Wexler and P.W. Culicover, *Formal Principles of Language Acquisition*, MIT Press, Cambridge (MA), 1980.

[35] R. Wiehagen, Limes-Erkennung rekursiver Funktionen durch spezielle Strategien, *Journal of Information Processing and Cybernetics* **12** (1976), 93–99. doi:10.1016/0306-4573(76)90031-5.

[36] R. Wiehagen, A thesis in inductive inference, in: *Nonmonotonic and Inductive Logic*, 1991, pp. 184–207. doi:10.1007/BFb0023324.

[37] T. Zeugmann and S. Lange, A guided tour across the boundaries of learning recursive languages, in: *Algorithmic Learning for Knowledge-Based Systems, GOSLER Final Report*, 1995, pp. 961:190–961:258.