On Learning Languages from Positive Data and a Limited Number of Short Counterexamples

Sanjay Jain[1,⋆] and Efim Kinber[2]

[1] School of Computing, National University of Singapore, Singapore 117543
sanjay@comp.nus.edu.sg

[2] Department of Computer Science, Sacred Heart University, Fairfield, CT 06432-1000, USA
kinbere@sacredheart.edu

Abstract. We consider two variants of a model for learning languages in the limit from positive data and a limited number of short negative counterexamples (counterexamples are considered to be short if they are smaller that the largest element of input seen so far). Negative counterexamples to a conjecture are examples which belong to the conjectured language but do not belong to the input language. Within this framework, we explore how/when learners using n short (arbitrary) negative counterexamples can be simulated (or simulate) using least short counterexamples or just 'no' answers from a teacher. We also study how a limited number of short counterexamples fairs against unconstrained counterexamples. A surprising result is that just one short counterexample (if present) can sometimes be more useful than any bounded number of counterexamples of least size. Most of results exhibit salient examples of languages learnable or not learnable within corresponding variants of our models.

1 Introduction

Our goal in this paper is to explore how limited amount of negative data, relatively easily available from a teacher, can help learning languages in the limit. There is a long tradition of using two popular different paradigms for exploring learning languages in the limit. One paradigm, learning languages from full positive data (all correct statements of the language), was introduced by Gold in his classical paper [Gol67]. In this model, **TxtEx**, the learner stabilizes in the limit to a grammar generating the target language. In another popular variant of this model, **TxtBc**, defined in [CL82] and [OW82] (see also [Bār74] and [CS83]) almost all conjectures outputted by the learner are correct grammars describing the target language. The second popular paradigm, learning using queries to a teacher (oracle) was introduced by D. Angluin in [Ang88]. In particular, D. Angluin considered three types of queries: subset, superset, and equivalence queries — when a learner asks if a current hypothesis generates a subset or a superset of the target language, or, respectively, generates exactly the target

⋆ Supported in part by NUS grant number R252-000-127-112.

G. Lugosi and H.U. Simon (Eds.): COLT 2006, LNAI 4005, pp. 259–273, 2006.

language. If the answer is negative, the teacher may provide a *counterexample* showing where the current hypothesis errs. This model has been used for exploring language learning primarily in the situation when no data was available in advance (see, for example, [LZ04b], [LZ04a]). In [JK06b], the two models were combined together: a learner gets full positive data and can query the teacher if the current conjecture is correct. On one hand, this model reflects the fact that a learner, during a process of acquisition of a new language, potentially gets access to all correct statements. On the other hand, this model adds another important tool, typically available, say, to a child learning a new language: a possibility to communicate with a teacher. Sometimes, this possibility may be really vital for successful learning. For example, if a learner of English past tense, having received on the input "call – called", "fall – fell", infers the rule implying that both past tense forms "called, cell" and "falled, fell" are possible, then this rule can be refuted only by counterexamples from a teacher.

In this context, subset queries are of primary interest, as they provide *negative counterexamples* if the learner errs, while other types of queries may provide positive 'counterexamples' eventually available on the input anyway (still, as it was shown in [JK06a], the sequel paper to [JK06b], superset and equivalence queries can make some difference even in presense of full positive data). Consequently, one can consider the learner for **NCEx** model as defined in [JK06b] (and its variant **NCBc** corresponding to **TxtBc** — **NC** here stands for 'negative counterexamples'), as making a subset query for each of its conjectures. When a learner tests every conjecture, potentially he/she can get indefinite number of counterexamples (still this number is, of course, finite if the learner learns the target language in the limit correctly). In [JK06a] the authors explored learning from positive data and *bounded* amount of additional negative data. In this context, one can consider three different scenarios of how subset queries and corresponding negative counterexamples (if any) can be used:

— only a bounded number (up to n) of subset queries is allowed during the learning process; this model was considered in [JK06a] under the name $\mathbf{SubQ}^n$;

— the learner makes subset query for every conjecture until n negative answers have been received; that is, the learner can ask potentially indefinite number of questions (however, still finite if the learning process eventually gives a correct grammar), but he is *charged* only when receiving a negative answer; this model was considered in [JK06a] under the name $\mathbf{NC}^n$;

— the learner makes subset queries for conjectures, when deemed necessary, until n negative answers have been received; in the sequel, we will refer to this model as $\mathbf{GNC}^n$, where **GNC** denotes 'generalized model of learning via negative counterexamples'.

Note that the $\mathbf{GNC}^n$ model combines the features of the first two (we have also demonstrated that it is stronger than each of the first two). All three models $\mathbf{SubQ}^n$, $\mathbf{NC}^n$, and $\mathbf{GNC}^n$ provide certain complexity measure (in the spirit of [GM98]) for learning languages that cannot be learned from positive data alone.

Negative counterexamples provided by the teacher in all these models are of arbitrary size. Some researchers in the field considered other types of negative data available for learners from full positive data. For example, negative data provided to learners in the model considered in [BCJ95] is preselected — in this situation just a very small amount of negative data can greatly enhance learning capabilities. A similar model was considered in [Mot91].

In this paper we explore models $\mathbf{SubQ}^n$, $\mathbf{NC}^n$, and $\mathbf{GNC}^n$ when the teacher provides a negative counterexample only if there is one whose size does not exceed the size of the longest statement seen so far. While learning from full positive data and negative counterexamples of arbitrary size can be interesting and insightful on its own right, providing arbitrary examples immediately (as it is assumed in the models under consideration) may be somewhat unrealistic — in fact, it may significantly slow down learning process, if not making it impossible. On the other hand, it is quite realistic to assume that the teacher can always reasonably quickly provide a counterexample (if any), if its size is bounded by the largest statement on the input seen so far. Following notation in [JK06a], we denote corresponding variants of our three models by $\mathbf{BSubQ}^n$, $\mathbf{BNC}^n$, and $\mathbf{BGNC}^n$, respectively. Following [Ang88] and [JK06a] we also consider restricted variants of the above three models — when the teacher, responding to a query, answers just 'no' if a counterexample of the size not exceeding the size of the largest statement seen so far exists — not providing the actual example; otherwise, the teacher answers 'yes'. To reflect this variant in the name of a model, we, following [JK06a], add the prefix **Res** to its name (for example, $\mathbf{ResBNC}^n$). It must be noted that, as it is shown in [JK06a], $\mathbf{BSubQ}^n$ does not provides any advantages over learning just from positive data. Therefore, we concentrate on $\mathbf{BNC}^n$, $\mathbf{BGNC}^n$ and their **Res** variants.

Our first goal in this research was to explore relationships between these two models as well as their restricted variants. Following [JK06b] and [JK06a], we also consider **Res** variants for models $\mathbf{NC}^n$, and $\mathbf{GNC}^n$ as well as their variants when the least (rather than arbitrary) counterexample is provided — in this case we use the prefix **L** (for example, $\mathbf{LNC}^n$). Consequently, we explore relationships between **B**-models and models using limited number of queries (including those getting just answers 'yes' or 'no'), or limited number of arbitrary or least counterexamples, or just answers 'no'. In this context, we, in particular, demonstrate advantages that our **B**-variants of learning (even **ResB**) can have over $\mathbf{GNC}^n$ in terms of the number of mind changes needed to arrive to the right conjecture.

In the full version of the paper (see [JK05]), we give also a number of results relating to comparison of **GNC**-model with **NC** model and comparison of learning via limited number of short counterexamples and finite number of queries. Most of our results provide salient examples of classes learnable (or not learnable) within corresponding models.

The paper has the following structure. In Section 2 we introduce necessary notation and definitions needed for the rest of the paper. In particular, we define some variants of the classical Gold's model of learning from texts (positive data): **TxtEx** — when the learner stabilizes to a correct (or nearly correct) conjecture generating the target language, and **TxtBc** — its behaviorally correct counterpart.

In Section 3, for both major models of learnability in the limit, **TxtEx** and **TxtBc**, we define two variants of learning from positive data and a uniformly bounded number of counterexamples: $\mathbf{NC}^n$ and $\mathbf{GNC}^n$, where the learner makes subset queries and is 'charged' for every negative answer from a teacher. We then define the main models considered in this paper: $\mathbf{BNC}^n$ and $\mathbf{BGNC}^n$, as well as **ResB** variants of both. We also formally define the **L** variant for all these models.

In Section 4 we explore relationships between different bounded negative counterexample models. In particular, we study the following two problems: under which circumstances, (a) **B**-learners receiving just answers 'yes' or 'no' can simulate the learners receiving short (possibly, even least) counterexamples; (b) learners receiving arbitrary short counterexamples can simulate the ones receiving the least short counterexamples. First, we note that in all variants of the paradigms **TxtEx** and **TxtBc**, an $\mathbf{LBNC}^n$-learner can be always simulated by a $\mathbf{ResBNC}^{2n-1}$-learner: $2n-1$ 'no' answers are enough to simulate n explicit negative counterexamples (similar fact holds also for the $\mathbf{LBGNC}^n$-learners). Moreover, for the $\mathbf{Bc}^*$ type of learnability (when almost all conjectures contain any finite number of errors), the number $2n-1$ in the above result drops to n (Theorem 6; note that, for learning via limited number of arbitrary or least counterexamples, the number $2n-1$ could not be lowered even for $\mathbf{Bc}^*$-learners, as shown in [JK06a]). On the other hand, the number $2n-1$ of negative answers/counterexamples cannot be lowered for the learning types $\mathbf{Ex}^*$ (when any finite number of errors in the limiting correct conjecture) and $\mathbf{Bc}^m$ (when the number of errors in almost all conjectures is uniformly bounded by some m) for both tasks (a) and (b). Namely, there exist $\mathbf{LBNC}^n\mathbf{Ex}$-learnable classes of languages that cannot be learned by $\mathbf{BGNC}^{2n-2}\mathbf{Bc}^m$ or $\mathbf{BGNC}^{2n-2}\mathbf{Ex}^*$-learners (Theorem 4) and there exist $\mathbf{BNC}^n\mathbf{Ex}$-learnable classes that cannot be learned by $\mathbf{ResBGNC}^{2n-2}\mathbf{Bc}^m$ or $\mathbf{ResBGNC}^{2n-2}\mathbf{Ex}^*$-learners (Theorem 5). We also show that a $\mathbf{LBNCEx}^*$-learner can be always simulated by a **ResBNCBc**-learner — when the number of negative answers/counterexamples is unbounded.

In Section 5 we explore relationships between our models when the counterexamples considered are short or unconstrained. First, we demonstrate how short counterexamples can be of advantage over unconstrained ones while learning from positive data and a bounded number of counterexamples. One of our central — somewhat surprising — results is that sometimes one 'no' answer, just indicating that a short counterexample exists, can do more than any number n of arbitrary (or even least) counterexamples used by (the strongest) $\mathbf{LGNC}^n\mathbf{Bc}^*$-learners (Theorem 9). Note that the advantages of least examples/counterexamples in speeding up learning has been studied in other situations also, such as learning of non-erasing pattern languages ([WZ94]). However, in our model of **BNC**-learning versus **LNC**-learning, the **LNC**-learner does get least counterexamples, and **BNC** learner gets just a counterexample, if there exists one below the maximal positive data seen so far. This seems on the surface to hurt, as **BNC**-learner is likely to get less (negative) data. In fact, that is the case when we do not bound the number of counterexamples received. However, when we consider counting/bounding, there is a *charge* for every counterexample. Consequently, a **BNC**-learner is not being charged for (unnecessary) negative data, if it does not receive it. As a result, the possibility of getting negative data which are $\leq$ maximal positive data seen in the input so far can be turned to an advantage — in terms of cost of learning. This is what is exploited in getting this result. We also show that sometimes a $\mathbf{ResBNC}^1\mathbf{Ex}$-learner can use just one mind change (and one 'no' answer witnessing existence of a short counterexample) to learn classes of languages not learnable by any **GNCEx**-learner using any bounded number of mind changes and an unbounded (finite) number of arbitrary counterexamples (Theorem 10). On the other hand, least counterexamples used by **NC**-type learners make a difference: any **LBNCEx**-learner using at most m mind changes and any (unbounded) number of counterexamples can be simulated by a $\mathbf{LNC}^m$-learner using at most m mind changes and at most m least counterexamples.

2 Notation and Preliminaries

Any unexplained recursion theoretic notation is from [Rog67]. The symbol N denotes the set of natural numbers, $\{0, 1, 2, 3, \ldots\}$. Symbols $\emptyset$, $\subseteq$, $\subset$, $\supseteq$, and $\supset$ denote empty set, subset, proper subset, superset, and proper superset, respectively. Cardinality of a set S is denoted by $\mathrm{card}(S)$. I_m denotes the set $\{x \mid x \leq m\}$. The maximum and minimum of a set are denoted by $\max(\cdot), \min(\cdot)$, respectively, where $\max(\emptyset) = 0$ and $\min(\emptyset) = \infty$. $L_1 \mathbf{\Delta} L_2$ denotes the symmetric difference of L_1 and L_2, that is $L_1 \mathbf{\Delta} L_2 = (L_1 - L_2) \cup (L_2 - L_1)$. For a natural number a, we say that $L_1 =^a L_2$, iff $\mathrm{card}(L_1 \mathbf{\Delta} L_2) \leq a$. We say that $L_1 =^* L_2$, iff $\mathrm{card}(L_1 \mathbf{\Delta} L_2) < \infty$. Thus, we take $n < * < \infty$, for all $n \in N$. If $L_1 =^a L_2$, then we say that L_1 is an a-variant of L_2.

We let $\langle \cdot, \cdot \rangle$ stand for an arbitrary, computable, bijective mapping from $N \times N$ onto N [Rog67]. We assume without loss of generality that $\langle \cdot, \cdot \rangle$ is monotonically increasing in both of its arguments. We define $\pi_1(\langle x, y \rangle) = x$ and $\pi_2(\langle x, y \rangle) = y$. We can extend pairing function to multiple arguments by using $\langle i_1, i_2, \ldots, i_k \rangle = \langle i_1, \langle i_2, \langle \ldots, \langle i_{k-1}, i_k \rangle \rangle \rangle \rangle$.

We let $\{W_i\}_{i \in N}$ denote an acceptable numbering of all r.e. sets. Symbol $\mathcal{E}$ will denote the set of all r.e. languages. Symbol L, with or without decorations, ranges over $\mathcal{E}$. By $\overline{L}$, we denote the complement of L, that is $N - L$. Symbol $\mathcal{L}$, with or without decorations, ranges over subsets of $\mathcal{E}$. By $W_{i,s}$ we denote the set W_i enumerated within s steps, in some standard computable method of enumerating W_i.

We now present concepts from language learning theory. A *sequence* σ is a mapping from an initial segment of N into $(N \cup \{\#\})$. Intuitively, #'s represent pauses in the presentation of data. The empty sequence is denoted by Λ. The *content* of a sequence σ, denoted $\mathrm{content}(\sigma)$, is the set of natural numbers in the range of σ. The *length* of σ, denoted by $|\sigma|$, is the number of elements in σ.

So, $|\Lambda| = 0$. For $n \leq |\sigma|$, the initial sequence of σ of length n is denoted by $\sigma[n]$. So, $\sigma[0]$ is Λ. We let σ, τ, and γ, with or without decorations, range over finite sequences. We denote the sequence formed by the concatenation of τ at the end of σ by $\sigma\tau$. SEQ denotes the set of all finite sequences.

A *text* T (see [Gol67]) for a language L is a mapping from N into $(N \cup \{\#\})$ such that L is the set of natural numbers in the range of T. $T(i)$ represents the $(i+1)$-th element in the text. The *content* of a text T, denoted by content(T), is the set of natural numbers in the range of T; that is, the language which T is a text for. $T[n]$ denotes the finite initial sequence of T with length n.

A *language learning machine from texts* (see [Gol67]) is an algorithmic device which computes a mapping from SEQ into N. We let $\mathbf{M}$, with or without decorations, range over learning machines. $\mathbf{M}(T[n])$ is interpreted as the grammar (index for an accepting program) conjectured by the learning machine $\mathbf{M}$ on the initial sequence $T[n]$. We say that $\mathbf{M}$ converges on T to i, (written: $\mathbf{M}(T)\!\downarrow = i$) iff $(\forall^\infty n)[\mathbf{M}(T[n]) = i]$.

There are several criteria for a learning machine to be successful on a language. Below we define some of them. All of the criteria defined below are variants of the **Ex**-style and **Bc**-style learning described in the Introduction; in addition, they allow a finite number of errors in almost all conjectures (uniformly bounded, or arbitrary). **TxtEx**-criteria is due to [Gol67]. $\mathbf{TxtEx}^a$ (for $a > 0$), and $\mathbf{TxtBc}^a$-criteria are due to [CL82]. Osherson and Weinstein [OW82] independently considered **TxtBc**.

Suppose $a \in N \cup \{*\}$. $\mathbf{M}$ $\mathbf{TxtEx}^a$*-identifies a language* L (written: $L \in \mathbf{TxtEx}^a(\mathbf{M})$) just in case for all texts T for L, $(\exists i \mid W_i =^a L)$ $(\forall^\infty n)[\mathbf{M}(T[n]) = i]$. $\mathbf{M}$ $\mathbf{TxtEx}^a$*-identifies a class* $\mathcal{L}$ of r.e. languages (written: $\mathcal{L} \subseteq \mathbf{TxtEx}^a(\mathbf{M})$) just in case $\mathbf{M}$ $\mathbf{TxtEx}^a$-identifies each language from $\mathcal{L}$. $\mathbf{TxtEx}^a = \{\mathcal{L} \subseteq \mathcal{E} \mid (\exists \mathbf{M})[\mathcal{L} \subseteq \mathbf{TxtEx}^a(\mathbf{M})]\}$.

$\mathbf{M}$ $\mathbf{TxtBc}^a$*-identifies an r.e. language* L (written: $L \in \mathbf{TxtBc}^a(\mathbf{M})$) just in case, for each text T for L, for all but finitely many n, $W_{\mathbf{M}(T[n])} =^a L$. $\mathbf{M}$ $\mathbf{TxtBc}^a$*-identifies a class* $\mathcal{L}$ of r.e. languages (written: $\mathcal{L} \subseteq \mathbf{TxtBc}^a(\mathbf{M})$) just in case $\mathbf{M}$ $\mathbf{TxtBc}^a$-identifies each language from $\mathcal{L}$. $\mathbf{TxtBc}^a = \{\mathcal{L} \subseteq \mathcal{E} \mid (\exists \mathbf{M})[\mathcal{L} \subseteq \mathbf{TxtBc}^a(\mathbf{M})]\}$. For $a = 0$, we often write **TxtEx** and **TxtBc**, instead of $\mathbf{TxtEx}^0$ and $\mathbf{TxtBc}^0$, respectively.

The following proposition is useful in proving many of our results.

Proposition 1. *[Gol67] Suppose L is an infinite language, $S \subseteq L$, and $L - S$ is infinite. Let $C_0 \subseteq C_1 \subseteq \cdots$ be an infinite sequence of finite sets such that $\bigcup_{i \in N} C_i = L$. Then $\{L\} \cup \{S \cup C_i \mid i \in N\}$ is not in* $\mathbf{TxtBc}^*$.

We let CYL_i denote the language $\{\langle i, x\rangle \mid x \in N\}$.

3 Learning with Negative Counterexamples to Conjectures

In this section we define two models of learning languages from positive data and negative counterexamples to conjectures. Both models are based on the general idea of learning from positive data and subset queries for the conjectures.

Intuitively, for learning with negative counterexamples to conjectures, we may consider the learner being provided a text, one element at a time, along with a negative counterexample to the latest conjecture, if any. (One may view this counterexample as a response of the teacher to the subset query when it is tested if the language generated by the conjecture is a subset of the target language). One may model the list of counterexamples as a second text for negative counterexamples being provided to the learner. Thus the learning machines get as input two texts, one for positive data, and other for negative counterexamples.

We say that $\mathbf{M}(T, T')$ converges to a grammar i, iff for all but finitely many n, $\mathbf{M}(T[n], T'[n]) = i$.

First, we define the basic model of learning from positive data and negative counterexamples to conjectures. In this model, if a conjecture contains elements not in the target language, then a counterexample is provided to the learner. **NC** in the definition below stands for 'negative counterexample'.

Definition 1. [JK06b] Suppose $a \in N \cup \{*\}$.

(a) $\mathbf{M}$ $\mathbf{NCEx}^a$*-identifies a language* L (written: $L \in \mathbf{NCEx}^a(\mathbf{M})$) iff for all texts T for L, and for all T' satisfying the condition:

$$T'(n) \in S_n, \text{ if } S_n \neq \emptyset \text{ and } T'(n) = \#, \text{ if } S_n = \emptyset,$$
$$\text{where } S_n = \overline{L} \cap W_{\mathbf{M}(T[n], T'[n])}$$

$\mathbf{M}(T, T')$ converges to a grammar i such that $W_i =^a L$.

(b) $\mathbf{M}$ $\mathbf{NCEx}^a$*-identifies* a class $\mathcal{L}$ of languages (written: $\mathcal{L} \subseteq \mathbf{NCEx}^a(\mathbf{M})$), iff $\mathbf{M}$ $\mathbf{NCEx}^a$-identifies each language in the class.

(c) $\mathbf{NCEx}^a = \{\mathcal{L} \mid (\exists \mathbf{M})[\mathcal{L} \subseteq \mathbf{NCEx}^a(\mathbf{M})]\}$.

For $\mathbf{LNCEx}^a$ criteria of inference, we consider providing the learner with the least counterexample rather than an arbitrary one. The criteria $\mathbf{LNCEx}^a$ of learning can thus be defined similarly to $\mathbf{NCEx}^a$, by requiring $T'(n) = \min(S_n)$, if $S_n \neq \emptyset$ and $T'(n) = \#$, if $S_n = \emptyset$ in clause (a) above (instead of $T'(n)$ being an arbitrary member of S_n).

Similarly, one can define $\mathbf{ResNCEx}^a$, where the learner is just told that the latest conjecture is or is not a subset of the input language, but is not provided any counterexamples in the case of 'no' answer.

For $\mathbf{BNCEx}^a$ criteria of inference, we update the definition of S_n in clause (a) of the definition of $\mathbf{NCEx}^a$-identification as follows: $S_n = \overline{L} \cap W_{\mathbf{M}(T[n], T'[n])} \cap \{x \mid x \leq \max(\text{content}(T[n]))\}$.

We can similarly define the criteria of inference $\mathbf{ResBNCEx}^a$, and $\mathbf{LBNCEx}^a$, $\mathbf{NCBc}^a$, $\mathbf{LNCBc}^a$, $\mathbf{ResBc}^a$, $\mathbf{BNCBc}^a$, $\mathbf{ResBNCBc}^a$ and $\mathbf{LBNCBc}^a$. We refer the reader to [JK06b] for more details, discussion and results about the various variations of **NCI**-criteria.

For $m \in N$, one may also consider the model, $\mathbf{NC}^m\mathbf{I}$, where, for learning a language L, the **NCI** learner is provided counterexamples only for its first m conjectures which are not subsets of L. For remaining conjectures, the answer provided is always $\#$. In other words, the learner is 'charged' only for the first m negative counterexamples, and the subset queries for later conjectures are not answered. Following is the formal definition.

Definition 2. [JK06a] Suppose $a \in N \cup \{*\}$, and $m \in N$.

(a) $\mathbf{M}$ $\mathbf{NC}^m\mathbf{Ex}^a$*-identifies a language* L (written: $L \in \mathbf{NC}^m\mathbf{Ex}^a(\mathbf{M})$) iff for all texts T for L, and for all T' satisfying the condition:

$$T'(n) \in S_n, \text{ if } S_n \neq \emptyset \text{ and } \mathrm{card}(\{r \mid r < n, T'(r) \neq \#\}) < m; T'(n) = \#,$$
$$\text{if } S_n = \emptyset \text{ or } \mathrm{card}(\{r \mid r < n, T'(r) \neq \#\}) \geq m,$$
$$\text{where } S_n = \overline{L} \cap W_{\mathbf{M}(T[n],T'[n])}$$

$\mathbf{M}(T, T')$ converges to a grammar i such that $W_i =^a L$.

(b) $\mathbf{M}$ $\mathbf{NC}^m\mathbf{Ex}^a$*-identifies* a class $\mathcal{L}$ of languages (written: $\mathcal{L} \subseteq \mathbf{NC}^m\mathbf{Ex}^a(\mathbf{M})$), iff $\mathbf{M}$ $\mathbf{NC}^m\mathbf{Ex}^a$-identifies each language in the class.

(c) $\mathbf{NC}^m\mathbf{Ex}^a = \{\mathcal{L} \mid (\exists \mathbf{M})[\mathcal{L} \subseteq \mathbf{NC}^m\mathbf{Ex}^a(\mathbf{M})]\}$.

For $a \in N \cup \{*\}$ and $\mathbf{I} \in \{\mathbf{Ex}^a, \mathbf{Bc}^a\}$, one can similarly define $\mathbf{BNC}^m\mathbf{I}$, $\mathbf{LBNC}^m\mathbf{I}$, $\mathbf{ResBNC}^m\mathbf{I}$ and $\mathbf{LNC}^m\mathbf{I}$, $\mathbf{ResNC}^m\mathbf{I}$ and $\mathbf{NC}^m\mathbf{Bc}^a$.

GNCI-identification model is same as the model of **NCI**-identification, except that counterexamples are provided to the learner only when it explicitly requests for such via a 'is this conjecture a subset of the target language' question (which we refer to as conjecture-subset question). This clearly does not make a difference if there is no bound on the number of questions asked resulting in counterexamples. However when there is a bound on number of counterexamples, then this may make a difference, as the **GNC**-learner may avoid getting a counterexample on some conjecture by not asking the conjecture-subset question. Thus, we will only deal with **GNC** model when there is a requirement of a bounded number of counterexamples. For $a \in N \cup \{*\}$ and $\mathbf{I} \in \{\mathbf{Ex}^a, \mathbf{Bc}^a\}$, one can define $\mathbf{GNC}^m\mathbf{I}$, $\mathbf{LGNC}^m\mathbf{I}$, $\mathbf{ResGNC}^m\mathbf{I}$ and $\mathbf{BGNC}^m\mathbf{I}$, $\mathbf{LBGNC}^m\mathbf{I}$, $\mathbf{ResBGNC}^m\mathbf{I}$, similarly to **NC** variants.

Note a subtle difference between models $\mathbf{LBGNC}^n$ and $\mathbf{LGNC}^n$: in the model $\mathbf{LBGNC}^n$, the teacher provides the shortest counterexample only if it is smaller than some element of the input, whereas there is no such requirement for $\mathbf{LGNC}^n$ (the same is true also for **NC**-variant).

4 Relations Among Bounded Negative Counterexample Models

In this section we establish relationships between **B**-variants of **NC** and **GNC**-models when any short, or the least short counterexamples, or just the 'no' answers about existence of short counterexamples are used.

First we establish that, similarly to the known result about **NC**-model ([JK06a]), number of counterexamples matters to the extent that $n+1$ 'no' answers used by **BNCEx**-style learners can sometimes do more that n least counterexamples obtained by $\mathbf{LBGNCBc}^*$-style learners.

Theorem 1. $\mathbf{ResBNC}^{n+1}\mathbf{Ex} - \mathbf{LBGNC}^n\mathbf{Bc}^* \neq \emptyset$.

The next result gives advantages of **GNC** model.

Theorem 2. *For all* $n, m \in N$, $\mathbf{ResBGNC}^1\mathbf{Ex} - (\mathbf{LBNC}^n\mathbf{Bc}^m \cup \mathbf{LBNC}^n\mathbf{Ex}^*) \neq \emptyset$.

Our main results in this section deal with the following problems: if and under which conditions, (a) **B**-learners receiving just 'yes' or 'no' answers can simulate learners receiving short (or, possibly, even least short) counterexamples, and (b) learners using arbitrary short counterexamples can simulate the ones receiving the least short counterexamples. We establish that, for both tasks (a) and (b), for the $\mathbf{Bc}^m$ and $\mathbf{Ex}^*$ types of learnability, $2n-1$ is the upper and the lower bound on the number of negative answers/examples needed for such a simulation. These results are similar to the corresponding results in [JK06a] for the model **NC**, however, there is also an interesting difference: as it will be shown below, for $\mathbf{Bc}^*$-learnability, the bound $2n-1$ can be lowered to just n (for $\mathbf{NCBc}^*$-learners, the lower bound $2n-1$ still holds).

First we establish the upper bound $2n-1$ for both tasks (a) and (b).

Theorem 3. *For all* $n \geq 1$,
(a) $\mathbf{LBNC}^n\mathbf{I} \subseteq \mathbf{ResBNC}^{2n-1}\mathbf{I}$.
(b) $\mathbf{LBGNC}^n\mathbf{I} \subseteq \mathbf{ResBGNC}^{2n-1}\mathbf{I}$.

Our next result shows that, for the $\mathbf{Bc}^m$ and $\mathbf{Ex}^*$ types of learnability, the bound $2n-1$ is tight in the strongest sense for the task (b). Namely, we show that **BNC**-learners using n least short counterexamples cannot be simulated by **BGNC**-learners using $2n-2$ (arbitrary short) counterexamples.

Theorem 4. *For all* $n \geq 1$, $\mathbf{LBNC}^n\mathbf{Ex} - (\mathbf{BGNC}^{2n-2}\mathbf{Bc}^m \cup \mathbf{BGNC}^{2n-2}\mathbf{Ex}^*) \neq \emptyset$.

Now we show that the bound $2n-1$ on the number of negative answers is tight for $\mathbf{Bc}^m$ and $\mathbf{Ex}^*$ types of learnability when **ResBNC**-learners try to simulate $\mathbf{BNC}^n$-learners.

Theorem 5. *For all* $m \in N$, $\mathbf{BNC}^n\mathbf{Ex} - (\mathbf{ResBGNC}^{2n-2}\mathbf{Bc}^m \cup \mathbf{ResBGNC}^{2n-2}\mathbf{Ex}^*) \neq \emptyset$.

Proof. Recall that $\langle x, y, z\rangle = \langle x, \langle y, z\rangle\rangle$. Thus, $\mathrm{CYL}_j = \{\langle j, x, y\rangle \mid x, y \in N\}$, and $\langle \cdot, \cdot, \cdot\rangle$ is increasing in all its arguments. Consider $\mathcal{L}$ defined as follows. For each $L \in \mathcal{L}$, there exists a set S, $\mathrm{card}(S) \leq n$, such that the conditions (1)–(3) hold.
(1) $L \subseteq \bigcup_{j\in S} \mathrm{CYL}_j$.
(2) $L \cap \mathrm{CYL}_j \cap \{\langle j, 0, x\rangle \mid x \in N\}$ contains exactly one element for each $j \in S$. Let this element be $\langle j, 0, \langle p_j, q_j\rangle\rangle$.
(3) For each $j \in S$,

(3.1) W_{p_j} is a grammar for $L \cap \mathrm{CYL}_j$ or
(3.2) $W_{p_j} \not\subseteq L$ and $W_{p_j} - L$ consists only of elements of form $\langle j, 1, 2x\rangle$ or only of elements of form $\langle j, 1, 2x+1\rangle$. Furthermore at least one such element is smaller than $\max(L)$. If this element is of form $\langle j, 1, 2z\rangle$, then $W_{q_j} = L \cap \mathrm{CYL}_j$. Otherwise, $L \cap \mathrm{CYL}_j$ is finite.

Intuitively, L may be considered as being divided into upto n parts, each part being subset of a cylinder, where each part satisfies the properties as given in (2) and (3).

Above class of languages can be seen to be in $\mathbf{BNC}^n\mathbf{Ex}$ as follows. On input σ, for each j such that content(σ) contains an element of CYL_j, find p_j and q_j as defined in condition 2 above (if σ does not contain any element of form $\langle j, 0, \langle p_j, q_j\rangle\rangle$, then grammar for $\emptyset$ is output on σ). Then for each of these j, learner computes a grammar for:

(a) W_{p_j} (if it has not received any counterexample from CYL_j),

(b) W_{q_j} (if the negative counterexample from CYL_j is of form $\langle j, 1, 2z\rangle$), and

(c) content$(T) \cap \mathrm{CYL}_j$, otherwise.

Then, the learner outputs a grammar for the union of the languages enumerated by the grammars computed for each j above. It is easy to verify that the above learner gets at most one counterexample from each CYL_j such that CYL_j intersects with the input language, and thus $\mathbf{BNC}^n\mathbf{Ex}$-identifies $\mathcal{L}$.

Proof of $\mathcal{L} \notin \mathbf{ResBGNC}^{2n-2}\mathbf{Bc}^m \cup \mathbf{ResBGNC}^{2n-2}\mathbf{Ex}^*$ is complex and we refer the reader to [JK05] for details. ∎

Interestingly, if we consider behaviorally correct learners that are allowed to make any finite number of errors in almost all correct conjectures, then n short (even least) counterexamples can be always substituted by just n 'no' answers. (For the model **NC**, the lower bound $2n - 1$ for the simulation by **Res**-type learners still holds even for $\mathbf{Bc}^*$-learnability, as shown in [JK06a]).

Theorem 6. *For all $n \in N$, $\mathbf{LBGNC}^n\mathbf{Bc}^* \subseteq \mathbf{ResBNC}^n\mathbf{Bc}^*$.*

Proof. First note that one can simulate a $\mathbf{LBGNC}^n\mathbf{Bc}^*$ learner $\mathbf{M}$ by a $\mathbf{LBNC}^n\mathbf{Bc}^*$ learner $\mathbf{M}'$ as follows. If $\mathbf{M}(\sigma, \sigma')$ does not ask a conjecture-subset question, then $\mathbf{M}'(\sigma, \sigma')$ is a grammar for $W_{\mathbf{M}(\sigma,\sigma')} - \{x \mid x \leq \max(\mathrm{content}(\sigma))\}$; otherwise $\mathbf{M}'(\sigma, \sigma') = \mathbf{M}(\sigma, \sigma')$. It is easy to verify that on any input text T, $\mathbf{M}'$ gets exactly the same counterexamples as $\mathbf{M}$ does, and all conjectures of $\mathbf{M}'$ are finite variants of corresponding conjectures of $\mathbf{M}$. Thus, any language $\mathbf{LBGNC}^n\mathbf{Bc}^*$-identified by $\mathbf{M}$ is $\mathbf{LBNC}^n\mathbf{Bc}^*$-identified by $\mathbf{M}'$.

Hence, it suffices to show that $\mathbf{LBNC}^n\mathbf{Bc}^* \subseteq \mathbf{ResBNC}^n\mathbf{Bc}^*$. Suppose $\mathbf{M}$ $\mathbf{LBNC}^n\mathbf{Bc}^*$-identifies $\mathcal{L}$. Define $\mathbf{M}'$ as follows. Suppose T is the input text.

The idea is for $\mathbf{M}'$ to output $\max(\mathrm{content}(T[m])) + 1$ variations of grammar output by $\mathbf{M}$ on $T[m]$. These grammars would be for the languages: $W_{\mathbf{M}(T[m])} - \{x \mid x \neq i \text{ and } x \leq \max(\mathrm{content}(T[m']))\}$, where $T[m']$ is the input seen by $\mathbf{M}'$ when generating this i-th variant (where $0 \leq i \leq \max(\mathrm{content}(T[m]))$). These grammars would thus allow $\mathbf{M}'$ to determine the least counterexample, if any, that the grammar output by $\mathbf{M}$ on $T[m]$ would have generated.

Formally conjectures of $\mathbf{M}'$ will be of form $P(j, m, i, s)$, where $W_{P(j,m,i,s)} = W_j - \{x \mid x \neq i \text{ and } x \leq s\}$.

We assume that $\mathbf{M}$ outputs $\emptyset$ until it sees at least one element in the input. This is to avoid having any counterexamples until input contains at least one element (which in turn makes the notation easier for the following proof).

On input $T[0]$, conjecture of $\mathbf{M}'$ is $P(\mathbf{M}(\Lambda, \Lambda), 0, 0, 0)$.

The invariants we will have is: If $\mathbf{M}'(T[m], T'[m]) = P(j, r, i, s)$, then, (i) $j = \mathbf{M}(T[r], T''[r])$, where $T''[r]$ is the sequence of least counterexamples for $\mathbf{M}$ on input $T[r]$ (for the language content(T)), (ii) $s = \max(\text{content}(T[m]))$, (iii) $r \leq m$, (iv) $i \leq \max(\text{content}(T[r]))$, and (v) $W_j - L$ does not contain any element $< i$. Invariants are clearly satisfied for $m = 0$.

Suppose $\mathbf{M}'(T[m], T'[m]) = P(\mathbf{M}(T[r], T''[r]), r, i, s)$. Then we define $\mathbf{M}'(T[m+1], T'[m+1])$ as follows.

If $T'(m)$ is 'no' answer, then let $T''(r) = i$, and let $\mathbf{M}'(T[m+1], T'[m+1]) = P(\mathbf{M}(T[r+1], T''[r+1]), r+1, 0, \max(\text{content}(T[m+1])))$.

Else if $i = \max(\text{content}(T[r]))$, then let $T''(r) = \#$, and let $\mathbf{M}'(T[m+1], T'[m+1]) = P(\mathbf{M}(T[r+1], T''[r+1]), r+1, 0, \max(\text{content}(T[m+1])))$.

Else, $\mathbf{M}'(T[m+1], T'[m+1]) = P(\mathbf{M}(T[r], T''[r]), r, i+1, \max(\text{content}(T[m+1])))$.

Now it is easy to verify that invariant is maintained. It also follows that T'' constructed as above is correct sequence of least counterexamples for $\mathbf{M}$ on input T. Moreover, each restricted 'no' answer in T' corresponds to a least counterexample in T''. Thus, $\mathbf{M}'$ gets exactly as many counterexamples as $\mathbf{M}$ does, and $\mathbf{M}'$ conjectures are $*$-variants of the conjectures of $\mathbf{M}$ (except that each conjecture of $\mathbf{M}$ is repeated finitely many times by $\mathbf{M}'$, with finite variations). It follows that $\mathbf{M}'$ $\mathbf{ResBNC}^n\mathbf{Bc}^*$-identifies $\mathcal{L}$. ∎

Corollary 1. *For all* $n \in N$, $\mathbf{LBNC}^n\mathbf{Bc}^* = \mathbf{BNC}^n\mathbf{Bc}^* = \mathbf{ResBNC}^n\mathbf{Bc}^* = \mathbf{LBGNC}^n\mathbf{Bc}^* = \mathbf{BGNC}^n\mathbf{Bc}^* = \mathbf{ResBGNC}^n\mathbf{Bc}^*$.

Our next result in this section shows how **BNCBc**-learners using just answers 'yes' or 'no' can simulate $\mathbf{LBNCEx}^*$-learners getting unbounded number of negative answers/counterexamples.

Proposition 2. $\mathbf{LBNCEx}^* \subseteq \mathbf{ResBNCBc}$.

We now consider error hierarchy for $\mathbf{BNC}^m$-learning model.

Theorem 7. *For all* $m, n \in N$,
(a) $\mathbf{TxtEx}^{2n+1} - \mathbf{LBGNC}^m\mathbf{Bc}^n \neq \emptyset$.
(b) $\mathbf{TxtEx}^{n+1} - \mathbf{LBGNC}^m\mathbf{Ex}^n \neq \emptyset$.
(c) For $\mathbf{I} \in \{\mathbf{ResBNC}^m, \mathbf{BNC}^m, \mathbf{LBNC}^m, \mathbf{ResBGNC}^m, \mathbf{BGNC}^m, \mathbf{LBGNC}^m\}$, $\mathbf{IEx}^{2n} \subseteq \mathbf{IBc}^n$.

5 Effects of Counterexamples Being Constrained/ Not-Constrained to Be Short

In this section we explore how, within the framework of our models, short counterexamples fair against arbitrary or least counterexamples (this includes also the cases when just answers 'no' are returned instead of counterexamples).

First, we use a result from [JK06a] to establish that one answer 'no' used by an **NCEx**-learner can sometimes do more than unbounded number of least (short) counterexamples used by $\mathbf{Bc}^*$-learners.

Theorem 8. *(based on [JK06a])* $\mathbf{ResNC^1Ex} - \mathbf{LBGNCBc^*} \neq \emptyset$.

From [JK06b] we have that, for $a \in N \cup \{*\}$, for $\mathbf{I} \in \{\mathbf{Ex}^a, \mathbf{Bc}^a\}$, $\mathbf{LBNCI} \subset \mathbf{ResNCI}$. Thus the next result is somewhat surprising. It shows that one short counterexample can sometimes give a learner more than any bounded number of least counterexamples. The proof exploits the fact that the learner is not charged if it does not get a counterexample.

Theorem 9. *For all $n \in N$,* $\mathbf{ResBNC^1Ex} - \mathbf{LGNC^nBc^*} \neq \emptyset$.

Proof. Let $A_k^j = \{\langle k, x\rangle \mid x \le j\}$. Let

$\mathcal{L} = \{L \mid (\exists S \mid \mathrm{card}(S) < \infty)(\exists f : S \rightarrow N)[$
1. $[k, k' \in S \ \wedge \ k < k'] \Rightarrow [\langle k, f(k)\rangle < \langle k', 0\rangle] \ \wedge$
2. $[L = \mathrm{CYL}_{\max(S)} \cup \bigcup_{k \in S - \max(S)} A_k^{f(k)}$ or
$L = \{\langle \max(S), f(\max(S)+2)\rangle\} \cup \bigcup_{k \in S} A_k^{f(k)}]\ ]\}$.

To see that $\mathcal{L} \in \mathbf{ResBNC^1Ex}$ consider the following learner. On input σ, if no 'no' answers are yet received, then the learner first computes $k = \max(\{j \mid \langle j, x\rangle \in \mathrm{content}(\sigma)\})$. Then it outputs a grammar for $L = \mathrm{CYL}_k \cup (\mathrm{content}(\sigma) - \mathrm{CYL}_k)$. If there is a 'no' answer which has been received, then the learner outputs a grammar for $\mathrm{content}(\sigma)$. It is easy to verify that the above learner $\mathbf{ResBNC^1Ex}$-identifies $\mathcal{L}$.

Now suppose by way of contradiction that some $\mathbf{M}$ $\mathbf{LGNC^nBc^*}$-identifies $\mathcal{L}$. Let $\sigma_0 = \sigma_0' = \Lambda, k_0 = 0$. Inductively define $\sigma_{i+1}, \sigma_{i+1}', f(k_i), k_{i+1}$ (for $i < n$) as follows.

Let σ be smallest extension of σ_i, if any, such that $\mathrm{content}(\sigma) \subseteq \mathrm{CYL}_{k_i} \cup \bigcup_{i'<i} A_{k_{i'}}^{f(k_{i'})}$ and $\mathbf{M}$ asks a conjecture-subset question on $(\sigma, \sigma_i' \#^{|\sigma|-|\sigma_i|})$ and $W_{\mathbf{M}(\sigma, \sigma_i' \#^{|\sigma|-|\sigma_i|})}$ contains an element which is not in $\mathrm{CYL}_{k_i} \cup \bigcup_{i'<i} A_{k_{i'}}^{f(k_{i'})}$ or is larger than $\max(\mathrm{content}(\sigma))$.

If there is such a σ, then let $\sigma_{i+1} = \sigma\#$, and $\sigma_{i+1}' = \sigma_i' \#^{|\sigma|-|\sigma_i|} w$ (where w is the least element in $W_{\mathbf{M}(\sigma, \sigma_i' \#^{|\sigma|-|\sigma_i|})}$ which is not in $\mathrm{CYL}_{k_i} \cup \bigcup_{i'<i} A_{k_{i'}}^{f(k_{i'})}$ or is larger than $\max(\mathrm{content}(\sigma))$). Let $f(k_i) = \max(\{y \mid \langle k_i, y\rangle \in \mathrm{content}(\sigma)\})$. Let k_{i+1} be such that $k_{i+1} > \langle k_i, f(k_i)\rangle$ and no element from $\mathrm{CYL}_{k_{i+1}}$ is present in $\mathrm{content}(\sigma_{i+1}')$.

Let m be largest value such that σ_m, σ_m' are defined above. Now, $\mathbf{M}$ has to $\mathbf{TxtBc^*}$-identify both $\mathrm{CYL}_{k_m} \cup \bigcup_{i<m} A_{k_m}^{f(k_m)}$ and $A_{k_m}^r \cup \{\langle k_m, r+2\rangle\} \cup \bigcup_{i<m} A_{k_i}^{f(k_i)}$, for all possible r, without any further counterexamples. An impossible task by Proposition 1. ∎

The above is the strongest possible result, as $\mathbf{ResNCI} \supseteq \mathbf{LBNCI}$ (see [JK06b]).

We now consider the complexity (mind change) advantages of having only short counterexamples. For this purpose, we need to modify the definition of learner slightly, to avoid biasing the number of mind changes. (This modification is used only for the rest of the current section).

Definition 3. A learner is a mapping from SEQ to $N \cup \{?\}$.

A learner $\mathbf{M}$ $\mathbf{TxtEx}_n$-identifies $\mathcal{L}$, iff it $\mathbf{TxtEx}$-identifies $\mathcal{L}$, and for all texts T for $L \in \mathcal{L}$, $\mathrm{card}(\{m \mid ? \neq \mathbf{M}(T[m]) \neq \mathbf{M}(T[m+1])\})$ is bounded by n.

One can similarly define the criteria with mind change bounds for learners receiving counterexamples. Our next result demonstrates that there exists a $\mathbf{TxtEx}$-learnable language (that is, learnable just from positive data — without any subset queries) that can be learned by a $\mathbf{BNC}^1\mathbf{Ex}$-learner using just one negative answer and at most one mind change and cannot be learned by $\mathbf{Ex}$-learners using any number of arbitrary counterexamples and any bounded number of mind changes.

Theorem 10. *There exists a $\mathcal{L}$ such that*

(a) $\mathcal{L} \in \mathbf{ResBNC}^1\mathbf{Ex}_1$.

(b) $\mathcal{L} \in \mathbf{TxtEx}$, and thus in $\mathbf{NCEx}$ and $\mathbf{GNCEx}$.

(c) For all m, $\mathcal{L} \notin \mathbf{GNCEx}_m$.

Proof. Let $L_n = \{x \mid x < n \text{ or } x = n+1\}$. Let $\mathcal{L} = \{L_n \mid n \in N\}$.

Consider the following learner. Initially output a grammar for N. If and when a 'no' answer is received, output a grammar for L_n, where n is the only counterexample received. It is easy to verify that above learner $\mathbf{ResBNC}^1\mathbf{Ex}_1$-identifies $\mathcal{L}$. Also, it is also easy to verify that $\mathcal{L} \in \mathbf{TxtEx}$ as one could output, in the limit on text T, a grammar for L_n, for the least n such that $n \notin \mathrm{content}(T)$.

We now show that $\mathcal{L} \notin \mathbf{NCEx}_m$. As the number of counterexamples are not bounded, it follows that $\mathcal{L} \notin \mathbf{GNCEx}_m$. Suppose by way of contradiction that $\mathbf{M}$ $\mathbf{NCEx}_m$-identifies $\mathcal{L}$. Then consider the following strategy to construct a diagonalizing language. We will construct the diagonalizing language in stages. Construction is non-effective. We will try to define l_s and u_s, and segments σ_s, σ'_s (σ'_s is the sequence of counterexamples), for $s \leq m+1$.

The following invariants will be satisfied.

(A) $u_s - l_s = 4^{m+3-s}$.

(B) $\mathbf{M}$ on proper prefixes of σ_s has made s different conjectures.

(C) $\mathrm{content}(\sigma_s) \subseteq \{x \mid x < l_s\}$.

(D) None of the conjectures made by $\mathbf{M}$ on proper prefixes of σ_s are for the language L_r, for $l_s \leq r \leq u_s$.

(E) $|\sigma'_s| = |\sigma_s|$.

(F) For $r < |\sigma_s|$, $\sigma'_s(r) = \#$, implies $W_{\mathbf{M}(\sigma_s[r], \sigma'_s[r])} \subseteq \{x \mid x < l_s\}$.

(G) For $r < |\sigma_s|$, $\sigma'_s(r) \neq \#$, implies $\sigma'_s(r) \in W_{\mathbf{M}(\sigma_s[r], \sigma'_s[r])}$, and $\sigma'_s(r) > u_s + 1$.

Initially, we let $l_0 = 0$ and $u_0 = l_0 + 4^{m+3}$, and $\sigma_0 = \sigma'_0 = \Lambda$. Note that invariants are satisfied.

Stage s (for $s = 0$ to $s = m$)

1. Let T be a text for L_{l_s} which extends σ_s.
2. Let $t \geq |\sigma_s|$, be the least value, if any, such that $\mathbf{M}(T[t], T'[t])$ is a conjecture different from any conjecture $\mathbf{M}(T[w], T'[w])$, for $w < |\sigma_s|$, where

$$T'(w) = \begin{cases} \sigma'_s(w), & \text{if } w < |\sigma_s|; \\ \#, & \text{if } w \geq |\sigma_s| \text{ and } \mathbf{M}(T[w], T'[w]) = ?; \\ T'(r), & \text{if } w \geq |\sigma_s| \text{ and } \mathbf{M}(T[w], T'[w]) = \mathbf{M}(T[r], T'[r]), \\ & \text{for some } r < |\sigma_s|. \end{cases}$$

(* Note that, in this step, we do not need the definition of $T'(w)$ when $\mathbf{M}(T[w], T'[w])$ makes a new conjecture at or beyond σ_s. For first such w (which is t found above) $T'(w)$ will be defined below). *)

If and when such a t is found, proceed to step 3.

3. Suppose $j = \mathbf{M}(T[t], T'[t])$.
If W_j contains an element $z \geq l_s + \frac{3(u_s - l_s)}{4}$, then
Let $l_{s+1} = l_s + \frac{u_s - l_s}{4}$.
Let $u_{s+1} = l_s + \frac{2(u_s - l_s)}{4}$.
Let $\sigma_{s+1} = T[t]\#$.
Let $\sigma'_{s+1} = T'[t]z$.
(* Note thus that $\mathbf{M}(T[t], T'[t])$ is not a correct grammar for L_r, where $l_{s+1} \leq r \leq u_{s+1}$. *)
Else,
Let $l_{s+1} = l_s + \frac{3(u_s - l_s)}{4}$.
Let $u_{s+1} = u_s$.
Let $\sigma_{s+1} = T[t]\#$.
Let $\sigma'_{s+1} = T'[t]\#$.
(* Note thus that $\mathbf{M}(T[t], T'[t])$ is not a correct grammar for L_r, where $l_{s+1} \leq r \leq u_{s+1}$. *)

End stage s

It is easy to verify that invariants are satisfied. (A) clearly holds by definition of l_{s+1} and u_{s+1} in step 3. (B) holds as one extra new conjecture is found at stage s, before proceeding to stage $s+1$. (C) holds, as $l_{s+1} \geq l_s + \frac{u_s - l_s}{4} > l_s + 2$, and content($T$) as defined in step 1 is a subset of L_{l_s}. (D) holds by induction, and noting that the conjecture at $T[t]$ as found in step 2 of stage s, is made explicitly wrong by appropriate choice of l_{s+1} and u_{s+1} in step 4. (E) easily holds by construction. (F) and (G) hold by the definition of σ'_{s+1} at step 3.

Now, if step 2 does not succeed at a stage $s \leq m$, then clearly $\mathbf{M}$ does not $\mathbf{NCEx}$-identify L_{l_s}. On the other hand if stage m does complete then $\mathbf{M}$ has already made $m+1$ different conjectures (and thus at least m mind changes) on prefixes of σ_{m+1}, which are not grammars for $L_{l_{m+1}}$. Thus, $\mathbf{M}$ cannot $\mathbf{NCEx}_m$-identify $L_{l_{m+1}}$. ∎

Let $X = \{x \mid x > 0\}$. If we consider the class $\mathcal{L} = \{L_n \mid n > 0\} \cup \{X\}$, then we can get the above result using *class preserving* learnability (that is, the learner always uses grammars from the numbering defining the target class of languages for its conjectures, see [ZL95] for formal definition) for $\mathbf{ResBNC}^1\mathbf{Ex}$.

Theorem 11. *For all $m \in N$, (a)* $\mathbf{LBNCEx}_m \subseteq \mathbf{LNCEx}_m$.
(b) $\mathbf{LBGNCEx}_m \subseteq \mathbf{LGNCEx}_m$.

References

[Ang88] D. Angluin. Queries and concept learning. *Machine Learning*, 2:319–342, 1988.

[Bār74] J. Bārzdiņš. Two theorems on the limiting synthesis of functions. In *Theory of Algorithms and Programs, vol. 1*, pages 82–88. Latvian State University, 1974. In Russian.

[BCJ95] G. Baliga, J. Case, and S. Jain. Language learning with some negative information. *Journal of Computer and System Sciences*, 51(5):273–285, 1995.

[CL82] J. Case and C. Lynes. Machine inductive inference and language identification. In M. Nielsen and E. M. Schmidt, editors, *Proceedings of the 9th International Colloquium on Automata, Languages and Programming*, volume 140 of *Lecture Notes in Computer Science*, pages 107–115. Springer-Verlag, 1982.

[CS83] J. Case and C. Smith. Comparison of identification criteria for machine inductive inference. *Theoretical Computer Science*, 25:193–220, 1983.

[GM98] W. Gasarch and G. Martin. *Bounded Queries in Recursion Theory.* Birkhauser, 1998.

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

[JK05] S. Jain and E. Kinber. On learning languages from positive data and a limited number of short counterexamples. Technical Report TR21/05, School of Computing, National University of Singapore, 2005.

[JK06a] S. Jain and E. Kinber. Learning languages from positive data and a finite number of queries. *Information and Computation*, 204:123–175, 2006.

[JK06b] S. Jain and E. Kinber. Learning languages from positive data and negative counterexamples. *Journal of Computer and System Sciences*, 2006. To appear.

[LZ04a] S. Lange and S. Zilles. Comparison of query learning and Gold-style learning in dependence of the hypothesis space. In Shai Ben-David, John Case, and Akira Maruoka, editors, *Algorithmic Learning Theory: Fifteenth International Conference (ALT' 2004)*, volume 3244 of *Lecture Notes in Artificial Intelligence*, pages 99–113. Springer-Verlag, 2004.

[LZ04b] S. Lange and S. Zilles. Replacing limit learners with equally powerful one-shot query learners. In John Shawe-Taylor and Yoram Singer, editors, *Proceedings of the Seventeenth Annual Conference on Learning Theory*, volume 3120 of *Lecture Notes in Artificial Intelligence*, pages 155–169. Springer-Verlag, 2004.

[Mot91] T. Motoki. Inductive inference from all positive and some negative data. *Information Processing Letters*, 39(4):177–182, 1991.

[OW82] D. Osherson and S. Weinstein. Criteria of language learning. *Information and Control*, 52:123–138, 1982.

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

[WZ94] R. Wiehagen and T. Zeugmann. Ignoring data may be the only way to learn efficiently. *Journal of Experimental and Theoretical Artificial Intelligence*, 6:131–144, 1994.

[ZL95] T. Zeugmann and S. Lange. A guided tour across the boundaries of learning recursive languages. In K. Jantke and S. Lange, editors, *Algorithmic Learning for Knowledge-Based Systems*, volume 961 of *Lecture Notes in Artificial Intelligence*, pages 190–258. Springer-Verlag, 1995.