NEGATIVE DATA IN LEARNING LANGUAGES

SANJAY JAIN

School of Computing, National University of Singapore, Singapore 117543.
Email: sanjay@comp.nus.edu.sg

EFIM KINBER

Department of Computer Science, Sacred Heart University, Fairfield, CT 06432-1000, U.S.A. Email: kinbere@sacredheart.edu

The paper is a survey of recent results on algorithmic learning (inductive inference) of languages from full collection of positive examples and some negative data. Different types of negative data are considered. We primarily concentrate on learning using (1) carefully chosen finite negative data (2) negative counterexamples provided when conjectures contain data not in the target language (3) negative counterexamples obtained from a teacher (formally, oracle), when a learner queries the oracle if an hypothesis is contained in the target language. We also explore how least counterexamples and counterexamples of bounded size fair against arbitrary counterexamples. The effects of random negative data are also briefly considered.

1. Introduction

Based on motivations from theories of language acquisition by children, Gold[10] developed an algorithmic model of learning (in the limit) from examples. This model may be described as follows. A learner receives as input, one by one, $x_0, x_1, \ldots$, where, $\{x_0, x_1, \ldots\}$ is exactly the target language, except possibly for a special pause symbol (which is useful for dealing with empty language). Note that there is no particular order among the elements $x_0, x_1, \ldots$, and repetitions are allowed. As the learner is receiving this data, it conjectures a sequence of grammars, $g_0, g_1, \ldots$ which are intended as descriptors of the target language. The learner can be regarded as successful if eventually the sequence of grammars stabilizes to a grammar g which generates/enumerates/accepts the target language. This model of learning is called **TxtEx** in the literature (**Txt** stands for "text", which is a complete positive data presentation, and "**Ex**" stands for explanatory learning). Note that it is more interesting to consider learnability of a

class of languages by a single learner (since, if we are only interested in learning one fixed language, then some learner — which just outputs the grammar for the fixed language — can easily learn it). The influence of Gold's paradigm to understanding human language learning is discussed in Pinker[24], Wexler and Culicover[28], Wexler[27] and Osherson, Stob and Weinstein[21].

Note that in the above model, the learner only receives elements of the language as input. It is not given any explicit information about elements not in the target language. This was based on the studies by linguists which hypothesised that children rarely, if ever, get negative information (see for example, Brown and Hanlon[5], Hirsh-Pasek, Treiman and Schneiderman[11] and Demetras, Post and Snow[8]).

Along with the above model of learning from positive data, Gold also studied learning from both positive and negative data. In this model, a learner is given all elements of the language, one by one, marked as positive, as well as all non-elements of the language, one by one, marked as negative. This criteria of learning is called **InfEx**. However based on studies about child learning, it is unrealistic to expect that children get all the negative data. On the other hand, as some studies point out, see Brown and Hanlon[5], Hirsh-Pasek, Treiman and Schneiderman[11] and Demetras, Post and Snow[8], children do get something more than just positive data.

The aim of the current paper is to survey some models of learning, where some amounts of negative data is provided to the learner. We will first consider two models of providing some *core* negative data to the learner. These models and results are based on work by Shinohara[26], Fulk [9], Motoki[18] and Baliga, Case and Jain[1]. We will then consider the case where negative data is provided to the learner via counterexamples to its conjectures. This is based on the philosophy that parents often correct their children by providing them counterexamples. This part is based on work done by the authors[13,12]. We also introduce and briefly consider a model in which learners are provided with random negative examples.

Before we study different models of negative data, it is useful to also consider some variants of the basic model of learnability from text as described above. Case and Lynes[6] (see also Osherson and Weinstein[22]) studied the case where the final hypothesis of the learner may not be accurate, but have upto n errors (finite number of errors). This criteria of learning is called $\mathbf{TxtEx}^n$ ($\mathbf{TxtEx}^*$). This was motivated by the fact that humans rarely, if ever, learn a language perfectly. Case and Lynes[6] (also see Osherson and Weinstein[22]) considered the case when learner need not syntactically

converge to a grammar, but eventually output only correct grammars (i.e., semantically converge rather than syntactically converge). In this model for all but finitely many n, the grammar g_n is a grammar for the target language. This criteria of learning is called **TxtBc**-learning. **Bc** here stands for behaviorally correct. $\mathbf{TxtBc}^n$ and $\mathbf{TxtBc}^*$ can be naturally defined.

Fulk[9] considered the case when, in addition to positive data, the learner is provided with a grammar for the complement of the language. Note that one can generate complete negative data using the grammar for the complement of the language. Fulk went on to show that this allows the learner to learn more than what can be learned using informants, that is using both complete positive and complete negative data. Though interesting, this model is quite unrealistic in the sense that children are definitely not given a grammar for the complement of the language. Most of the literature (see for example, Brown and Hanlon[5], Hirsh-Pasek, Treiman and Schneiderman[11] and Demetras, Post and Snow[8]) also argues that children do not get complete negative data. What is more realistic is that a learner is provided with some negative data, probably carefully selected or based on what the child has learnt (that is in a way based on child's current conjecture). Jain and Sharma[15] considered a modification where the learner instead of being given a grammar for the complete $\overline{L}$, is given only a grammar for a subset of $\overline{L}$, where this subset satisfies some density constraints. Despite being somewhat weaker than Fulk's model, it still seems unrealistic to expect that children are provided with grammars for any parts of the complement of the target language.

Based on this, Shinohara[26] considered the case where the learner is given $\geq n$ (n fixed beforehand) arbitrary negative examples along with the complete positive data about the language. Clearly, this is possible only when complement of the language does contain at least n elements. Shinohara showed that this method of presenting negative data is not useful, in the sense that it does not give any learnability advantages over just positive data. Extending this work, Baliga, Case and Jain[1] considered the case that the learner is given upto n carefully chosen elements of the complement of the language. These negative examples may be considered as *core* negative data. Intuitively, this was aimed to model the situation when a teacher carefully selects the negative examples to be provided to the student. Indeed, as expected this model turned out to be quite powerful. For example, it can be shown that the class of all recursively enumerable sets, $\mathcal{E}$, can be learned by some learner in **TxtEx** sense, when it additionally receives upto two carefully selected negative examples. Even one carefully

selected negative example is enough if one allows upto one error in the final grammar, or allows behaviourally correct learning. In contrast one carefully selected negative example is not enough to learn the class $\mathcal{E}$ according to **TxtEx** crtieria, though it still can be shown to be quite useful.

The reason for this apparent gains by having only one or two negative examples in the above model is based on the fact that one can "code" information into these negative data, allowing the learner to essentially extract a grammar for the target language from the negative data. To avoid such coding, Baliga, Case and Jain (motivated by a model considered by Motoki[18]) considered the following modification. For each possible target language, besides the core negative data, the learner may be given some further negative data. This model of learning is called open negative data, reminding one of the basic open sets for the topology with respect to which enumeration operators are continuous. As the learner may not be able to distinguish core negative data from the other negative data, the effects of "coding" are somewhat eliminated. This model turned out to be quite useful in studying the effects of negative data. In particular, above criteria lie strictly between **TxtEx** and **InfEx** models of learning. Let $\mathbf{NegO}^n\mathbf{I}$ ($\mathbf{NegO}^*\mathbf{I}$) denote the criteria of learning formed when the core negative information is of size at most n (the core negative information is of finite size), and **I** is the basic model of learning (such as $\mathbf{Ex}^a$, or $\mathbf{Bc}^a$). It can be shown that $\mathbf{NegO}^*\mathbf{I}$ turns out to be of the same power as **InfI**. Furthermore, each additional element allowed in the core, gives learnability advantages (that is $\mathbf{NegO}^{n+1}\mathbf{I}$ allows learning strictly more classes compared to $\mathbf{NegO}^n\mathbf{I}$). On the other hand, the finite negative core information is not enough to overcome extra errors (that is, one can learn something in $\mathbf{TxtEx}^{n+1}$ model of learning, but cannot in $\mathbf{NegO}^*\mathbf{TxtEx}^n$ model of learning). Additionally, it was shown that small packets of negative information also lead to increased *speed* of learning. This result agrees with a psycholinguistic hypothesis of McNeill correlating the availability of parental expansions with the speed of child language development. McNeill[17] posits that there is *faster* learning of language for children in homes in which more corrections (usually in the form of *possibly exemplary* expansions) are given. These corrections are, in part, a form of negative information.

Note that in both models considered above, one selects carefully negative examples based on the language being learned. However, in reality often negative examples are formed more as "counterexamples" based on errors done by child, rather than being preselected. To model such a situation, authors[13] considered a criteria of learning where the learner is given a

negative counterexample to each of its conjectures, if it exists. This model of learning is called **NCEx**. This model turned out to be robust with respect to different variations (giving least counterexamples, or the counterexamples being delayed). Besides the usual hierarchy results showing the advantages of having counterexamples, the paper[13] contrasts this criteria with **TxtEx** and **InfEx**, showing that in some cases structurally it behaves more like "**InfEx**" rather than like "**TxtEx**". For example, results such as (a) if $\mathcal{L} \in$ **NCEx** then so is $\mathcal{L} \cup \mathcal{S}$, for any finite class $\mathcal{S}$ of recursive languages, (b) **NCEx*** $\subseteq$ **NCBc** follow more along the lines of results in learning from informants. On the other hand, it is shown that in some cases full negative data, informant, is needed for learning, and just counterexamples are not enough. A surprising result, in the case of behaviorally correct learning is that the whole class $\mathcal{E}$ can be learned in **NCBc**[1] model — making it more powerful than even learning from informants! (by contrast **NCBc** $\subset$ **InfBc** and **NCEx**^{a} $\subseteq$ **InfEx**^{a}).

An interesting complexity aspect is that, for **Ex** model, though **NCEx** is a strict subset of **InfEx**, it can sometimes give huge complexity advantages. That is, in some cases one can learn a class in **NCEx** model using only n mind changes, whereas learning with informants requires exponentially many mind changes. In a variation of **NCEx** model, where least negative counterexamples are given, one can even show that there are classes which are learnable using 1 mind change, though learning with informants requires unbounded number of mind changes! Though, as mentioned above, several variations of negative counterexample models do not give different learning power, there is often complexity advantages which may result from a particular variation.

Learning from counterexamples also addresses a general concern about overgeneralization in learning. When one only receives positive data, then overgeneralized hypothesis cannot be corrected based on input data alone. However, if negative counterexamples are provided to the learner, then one can address this issue.

One can view getting counterexamples, as asking a "subset query" about the conjecture to a teacher. However in the usual model of learning from subset queries, a learner is allowed to query about other languages (besides just the conjectured language) being subsets of target language. This led us[12] to consider learning with subset (and other kind of) queries. It can be shown that if a **TxtEx** learner is allowed finitely many (but unbounded) subset queries, then the learning ability is same as that in the **NCEx** model. If the learner is allowed infinitely many subset queries, then a

learner (using texts) can learn all the recursively enumerable languages. Thus it is more interesting to study the case when the number of queries is bounded. Authors showed several results comparing the criteria of learning with negative counterexamples and subset queries, and giving hierarchies based on number of queries allowed. They also showed hierarchies based on variations of the query model where no answers are accompanied by least, arbitrary, or no counterexamples.

An interesting research work to consider would be to see how random negative examples work — this may be more closer to how humans learn languages. It can be shown that often random negative examples do help.

2. Preliminaries

2.1. *Notation*

Any unexplained recursion theoretic notation is from Rogers[25]. N denotes the set of natural numbers, $\{0, 1, 2, 3, \ldots\}$. $*$ denotes a non-member of N and is assumed to satisfy $(\forall n)[n < * < \infty]$. $\emptyset$ denotes the empty set. $\subseteq$, $\subset$, $\supseteq$ and $\supset$ respectively denote subset, proper subset, superset, and proper superset. $\text{card}(S)$ denotes the cardinality of S. $S_1 =^n S_2$ denotes $\text{card}((S_1 - S_2) \cup (S_2 - S_1)) \leq n$; $S_1 =^* S_2$ means that $\text{card}((S_1 - S_2) \cup (S_2 - S_1))$ is finite. $\downarrow$ denotes defined and $\uparrow$ denotes undefined. $\max(\cdot), \min(\cdot)$ denote the maximum and minimum of a set, respectively, where $\max(\emptyset) = 0$ and $\min(\emptyset) = \uparrow$. $\langle i, j \rangle$ stands for an arbitrary, computable, one-to-one encoding of all pairs of natural numbers onto N (see for example[25]).

The quantifiers '$\overset{\infty}{\forall}$', and '$\overset{\infty}{\exists}$' essentially from Blum[4], mean 'for all but finitely many' and 'there exist infinitely many', respectively. The quantifier '$\exists!$' means 'there exists a unique'.

φ denotes a fixed *acceptable* programming system for the partial computable functions: $N \rightarrow N$ (see the books[25,16]). φ_i denotes the partial computable function computed by program i in the φ-system.

W_i denotes $\text{domain}(\varphi_i)$. W_i is, then, the r.e. set/language ($\subseteq N$) accepted (or equivalently, generated) by the φ-program i. $\mathcal{E}$ will denote the set of all r.e. languages. L, with or without decorations, ranges over $\mathcal{E}$. $\overline{L}$ denotes the complement of L. $\mathcal{L}$, with or without decorations, ranges over subsets of $\mathcal{E}$. $\mathcal{L} = \{L_i \mid i \in N\}$ is called an indexed family iff there exist a recursive function f such that $f(i, x) = 1$ iff $x \in L_i$.

2.2. *Some Notions from Language Learning*

We now consider some basic notions in language learning. Following definition gives the concepts of data that is presented to a learner. Part (a) considers the notion of positive data, and part (b) considers the case when both positive and negative data are given.

Definition 2.1. (Gold[10])

(a) A *text* T is a mapping from N into $(N \cup \{\#\})$. The *content* of a text T, denoted content(T), is the set of natural numbers in the range of T.

(b) An infinite information sequence I is a mapping from N to $(N \times \{0,1\}) \cup \{\#\}$, such that if (x,b) appears in the sequence, then $(x,1-b)$ does not appear in the sequence. The *content* of an information sequence I denoted content(I), is the set of pairs in the range of I. PosInfo(I) $= \{x \mid (x,1) \in \text{content}(I)\}$, and NegInfo($I$) $= \{x \mid (x,0) \in \text{content}(I)\}$.

(c) T is a text for L iff content(T) $= L$. I is an information sequence for L iff PosInfo(I) $= L$ and NegInfo(I) $= \overline{L}$.

(d) $T[n]$ denotes the initial segment of T of length n. Similarly, $I[n]$ denotes the initial segment of I of length n.

We let T (I), with or without superscripts, range over texts (information sequences).

Intuitively, #'s in the texts/information sequences denote pauses in the presentation of data. For example, the only text for the empty language is just an infinite sequence of #'s. Note that by our convention on information sequences, PosInfo(I) $\cap$ NegInfo(I) $= \emptyset$.

A finite sequence σ is an initial segment of a text or an infinite information sequence. One can similarly define content(σ) (and PosInfo(σ), NegInfo(σ) in case of σ being initial segment of an information sequence).

SEQ denotes the set of all finite initial segments of texts. SEG denotes the set of all finite initial segments of information sequences. Note that SEQ and SEG can be coded onto N.

Definition 2.2. A *language learning machine* is an algorithmic device which computes a mapping from SEQ (or SEG) into N.

Later we will consider variation of learning machines. For convenience of exposition we avoid defining these variants until we need them.

We let $\mathbf{M}$, with or without decorations, range over learning machines. We say that $\mathbf{M}(T)\downarrow = i \Leftrightarrow (\overset{\infty}{\forall}\ n)[\mathbf{M}(T[n]) = i]$. Convergence on information sequences is similarly defined.

We now define some common criteria for learning. Our first criterion is based on learner, given a text for the language, converging to a grammar for the language.

Definition 2.3. (Gold[10], Case and Lynes[6], Osherson and Weinstein[22]) Let $a \in N \cup \{*\}$.

(a) $\mathbf{M}$ $\mathbf{TxtEx}^a$-*identifies* L (written: $L \in \mathbf{TxtEx}^a(\mathbf{M})$) $\Leftrightarrow$ ($\forall$ texts T for L)($\exists i \mid W_i =^a L$)$[\mathbf{M}(T)\downarrow = i]$.

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

The criterion we call $\mathbf{TxtEx}^0$ is due to Gold[10]. The $a > 0$ case is from Case and Lynes[6] (Osherson and Weinstein[22] independently introduced the $a = *$ case). We refer the reader to Pinker[24], Wexler and Culicover[28], Wexler[27], Osherson, Stob, and Weinstein[19,20,21], and Jain *et al*[14] for further discussion on the paradigm.

The next definition is based on learner semantically rather than syntactically converging to the grammar(s) for the language.

Definition 2.4. (Case and Lynes[6]) Let $a \in N \cup \{*\}$.

(a) $\mathbf{M}$ $\mathbf{TxtBc}^a$-*identifies* L (written: $L \in \mathbf{TxtBc}^a(\mathbf{M})$) $\Leftrightarrow$ ($\forall$ texts T for L)$(\overset{\infty}{\forall}\ n)[W_{\mathbf{M}(T[n])} =^a L]$.

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

The $a \in \{0, *\}$ cases were independently introduced by Osherson and Weinstein[22,23]. The corresponding notion in the case of learning functions was introduced by Bārzdiņš[2] and Case and Smith[7].

We now consider the corresponding learning criteria when information sequences are provided to the learner.

Definition 2.5. (Gold[10] and Case and Lynes[6] Let $a \in N \cup \{*\}$.

(a) $\mathbf{M}$ $\mathbf{InfEx}^a$-*identifies* L (written: $L \in \mathbf{InfEx}^a(\mathbf{M})$) $\Leftrightarrow$ for all information sequences I for L, $\mathbf{M}(I)\downarrow$ and $W_{\mathbf{M}(I)} =^a L$.

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

(b) $\mathbf{M}$ $\mathbf{InfBc}^a$-*identifies* L (written: $L \in \mathbf{InfBc}^a(\mathbf{M})$) $\Leftrightarrow$ for all information sequences I for L, $(\overset{\infty}{\forall}\ n)[W_{\mathbf{M}(I[n])} =^a L]$.

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

We often write **TxtEx** (respectively, **TxtBc**, **InfEx**, **InfBc**) for $\mathbf{TxtEx}^0$ (respectively, $\mathbf{TxtBc}^0$, $\mathbf{InfEx}^0$, $\mathbf{InfBc}^0$).

The following theorem gives some basic comparison between the criteria of inference discussed above. Note that by definition, for all $a \in N \cup \{*\}$, $\mathbf{TxtEx}^a \subseteq \mathbf{InfEx}^a \cap \mathbf{TxtBc}^a$, and $(\mathbf{TxtBc}^a \cup \mathbf{InfEx}^a) \subseteq \mathbf{InfBc}^a$.

Theorem 2.1. *(Gold[10], Blum and Blum[3], Case and Lynes[6] and Case and Smith[7]) For all $n \in N$, the following hold.*

(a) $\mathbf{TxtEx}^{n+1} - \mathbf{InfEx}^n \neq \emptyset$.
(b) $\mathbf{TxtEx}^* - \bigcup_{m \in N} \mathbf{InfEx}^m \neq \emptyset$.
(c) $\mathbf{TxtBc} - \mathbf{InfEx}^* \neq \emptyset$.
(d) $\mathbf{TxtBc}^{n+1} - \mathbf{InfBc}^n \neq \emptyset$.
(e) $\mathbf{TxtBc}^* - \bigcup_{m \in N} \mathbf{InfBc}^m \neq \emptyset$.
(f) $\mathbf{TxtEx}^{2n} \subset \mathbf{TxtBc}^n$.
(g) $\mathbf{TxtEx}^{2n+1} - \mathbf{TxtBc}^n \neq \emptyset$.
(h) $\mathbf{InfEx}^* \subseteq \mathbf{InfBc}^0$.
(i) $\mathbf{InfEx} - \mathbf{TxtBc}^* \neq \emptyset$.
(j) $\mathcal{E} \in \mathbf{InfBc}^*$.

3. Identification with Finite Negative Information

We first consider the model where an apparently small finite set of negative information is given in addition to text. In part (a) of both Definitions 3.1 and 3.2 just below, S is the *core* of negative information. The learner gets (besides the positive data) exactly this core negative data (marked as such) and no other negative data.

Definition 3.1. (Baliga, Case and Jain[1]) Suppose $a, b \in N \cup \{*\}$.

(a) $\mathbf{M}$ $\mathbf{NegF}^b\mathbf{TxtEx}^a$*-identifies* $L \in \mathcal{E}$ (written: $L \in \mathbf{NegF}^b\mathbf{TxtEx}^a(\mathbf{M})$) $\Leftrightarrow$ $(\exists S \subseteq \overline{L} \mid \mathrm{card}(S) \leq b)(\forall I \mid \mathrm{PosInfo}(I) = L \ \& \ \mathrm{NegInfo}(I) = S)[\mathbf{M}(I)\downarrow \text{ and } W_{\mathbf{M}(I)} =^a L]$.

(b) $\mathbf{NegF}^b\mathbf{TxtEx}^a = \{\mathcal{L} \subseteq \mathcal{E} \mid (\exists \mathbf{M})[\mathcal{L} \subseteq \mathbf{NegF}^b\mathbf{TxtEx}^a(\mathbf{M})]\}$.

Definition 3.2. (Baliga, Case and Jain[1]) Suppose $a, b \in N \cup \{*\}$.

(a) $\mathbf{M}$ $\mathbf{NegF}^b\mathbf{TxtBc}^a$*-identifies* $L \in \mathcal{E}$ (written: $L \in \mathbf{NegF}^b\mathbf{TxtBc}^a(\mathbf{M})$) $\Leftrightarrow$ $(\exists S \subseteq \overline{L} \mid \mathrm{card}(S) \leq b)(\forall I \mid \mathrm{PosInfo}(I) = L \ \& \ \mathrm{NegInfo}(I) = S)(\overset{\infty}{\forall}\, n)[W_{\mathbf{M}(I[n])} =^a L]$.

(b) $\mathbf{NegF}^b\mathbf{TxtBc}^a = \{\mathcal{L} \subseteq \mathcal{E} \mid (\exists \mathbf{M})[\mathcal{L} \subseteq \mathbf{NegF}^b\mathbf{TxtBc}^a(\mathbf{M})]\}$.

By definition, for all a, $\mathbf{NegF}^0\mathbf{TxtEx}^a = \mathbf{TxtEx}^a$ and $\mathbf{NegF}^0\mathbf{TxtBc}^a = \mathbf{TxtBc}^a$.

The next theorem illustrates the gain in learning power obtained by using sets of negative information with cardinality at most one/two.

Theorem 3.1. *(Baliga, Case and Jain[1]) $\mathcal{E} \in \mathbf{NegF}^2\mathbf{TxtEx} \cap \mathbf{NegF}^1\mathbf{TxtEx}^1 \cap \mathbf{NegF}^1\mathbf{TxtBc}$.*

In contrast to the above result, we have:

Theorem 3.2. *(Baliga, Case and Jain[1]) $\mathcal{E} \notin \mathbf{NegF}^1\mathbf{TxtEx}$.*

However $\mathbf{NegF}^1\mathbf{TxtEx}$ is still quite powerful as shown by the following theorem.

Theorem 3.3. *(Baliga, Case and Jain[1])*
(a) $\{L \in \mathcal{E} \mid \overline{L} \text{ is infinite}\} \in \mathbf{NegF}^1\mathbf{TxtEx}$.
(b) $\mathbf{NegF}^1\mathbf{TxtEx} - \mathbf{TxtBc}^* \neq \emptyset$.
(c) $\mathbf{NegF}^1\mathbf{TxtEx} - \mathbf{InfBc}^n \neq \emptyset$.
(d) $\mathbf{TxtEx}^1 \subset \mathbf{NegF}^1\mathbf{TxtEx}$.
(e) $\mathbf{InfEx} \subseteq \mathbf{NegF}^1\mathbf{TxtEx}$.

For $i \geq 2$, it is open at present whether $\mathbf{TxtEx}^i \subset \mathbf{NegF}^1\mathbf{TxtEx}$.

4. Some other Negative Information Models

Shinohara[26] considered giving to the learner atleast (but arbitrary) n negative data items.

Definition 4.1. (Shinohara[26]) Let $n \in N$.
(a) Suppose $\overline{L}$ has at least n elements. $\mathbf{M}$ $\mathbf{PP}^n$*-identifies* L (written $L \in \mathbf{PP}^n(\mathbf{M})$), iff for all information sequences I such that $\text{PosInfo}(I) = L$ and $\text{card}(\text{NegInfo}(I)) \geq n$, $\mathbf{M}(I)$ converges to a grammar for L.
(b) $\mathbf{PP}^n = \{\mathcal{L} \mid (\exists \mathbf{M})[\mathcal{L} \subseteq \mathbf{PP}^n(\mathbf{M})]\}$.

Theorem 4.1. *(Shinohara[26]) Let $n \in N$. Suppose for any $L \in \mathcal{L}$, $\overline{L}$ contains at least n elements. Then $\mathcal{L} \in \mathbf{TxtEx}$ iff $\mathcal{L} \in \mathbf{PP}^n$.*

Fulk considered giving the grammar for the complement of L to the learner. For this notion consider $\mathbf{M}$ as being given two inputs: (a) a grammar, and (b) a text. Convergence of $\mathbf{M}(i, T)$ can be defined as usual.

Definition 4.2. (Fulk[9])
(a) $\mathbf{M}$ $\mathbf{CTxtEx}$-identifies L (written: $L \in \mathbf{CTxtEx}(\mathbf{M})$) iff for all i such that $W_i = \overline{L}$, for all texts T for L, $\mathbf{M}(i, T)$ converges to a grammar for L.

(b) $\mathbf{CTxtEx} = \{\mathcal{L} \mid (\exists \mathbf{M})[\mathcal{L} \subseteq \mathbf{CTxtEx}(\mathbf{M})]\}$.

Fulk showed that having a grammar for the complement gives tremendous advantages.

Theorem 4.2. *(Fulk[9]) Let $n \in N$.* $\mathbf{CTxtEx} - \mathbf{InfBc}^n \neq \emptyset$.

Fulk also considered the case when instead of being given a grammar for complement of L, the learner is given a sequence of grammars all but finitely many of which are grammars for L. It is not known at present whether this gives any advantages over informants.

Jain and Sharma[15] considered giving a grammar for a subset of the complement of the language being learned, where this subset has certain density.

Motoki[18] considered a form of open negative information as follows.

Definition 4.3. (Motoki[18]) $\mathbf{M}$ *identifies L using advisor A_L* iff for all information sequences I such that $\text{PosInfo}(I) = L$ and $\text{NegInfo}(L) \supseteq A_L$, $\mathbf{M}(I)$ converges to a grammar for L.

We use a general definition, though Motoki was mainly interested in indexed families.

Motoki showed that there exists a class $\mathcal{L} \notin \mathbf{TxtEx}$, such that a learner $\mathbf{M}$ can identify each $L \in \mathcal{L}$ using some advisor A_L, where $\text{card}(A_L) \leq 1$. Motoki also gave a characterization of indexed families which can be learned using some advisor. We will be discussing a general form of open negative information in the next section.

5. Identification with Open Negative Information

We now consider another model of presenting negative information to learning machines. Here the negative information is supplied in a manner reminding one of the basic open sets for the topology with respect to which enumeration operators are continuous. This is the first topology described in Exercise 11–35, page 217 of Rogers[25]. These models were motivated in part by those considered by Motoki[18] (see Definition 4.3 above) and those in Section 3 above. Basically, this model allows the possibility of more negative information being supplied in addition to the finite cores of negative information.

Definition 5.1. (Baliga, Case and Jain[1]) Suppose $a, b \in N \cup \{*\}$.

(a) $\mathbf{M}$ $\mathbf{NegO}^b\mathbf{TxtEx}^a$-*identifies* $L \in \mathcal{E}$ (written: $L \in \mathbf{NegO}^b\mathbf{TxtEx}^a(\mathbf{M})$) $\Leftrightarrow$ $(\exists S \subseteq \overline{L} \mid \text{card}(S) \leq b)(\forall I \mid \text{PosInfo}(I) = L \ \& \ S \subseteq \text{NegInfo}(I) \subseteq \overline{L})[\mathbf{M}(T)\downarrow \text{ and } W_{\mathbf{M}(I)} =^a L]$.

(b) $\mathbf{NegO}^b\mathbf{TxtEx}^a = \{\mathcal{L} \subseteq \mathcal{E} \mid (\exists \mathbf{M})[\mathcal{L} \subseteq \mathbf{NegO}^b\mathbf{TxtEx}^a(\mathbf{M})]\}$.

Thus, in contrast with Definition 3.1, in above model the learner must satisfy the stronger constraint that it needs to learn when the negative information present in the data given to it is any S' such that $S \subseteq S' \subseteq \overline{L}$ (here S' may be infinite).

Definition 5.2. (Baliga, Case and Jain[1]) Suppose $a, b \in N \cup \{*\}$.

(a) $\mathbf{M}$ $\mathbf{NegO}^b\mathbf{TxtBc}^a$-*identifies* $L \in \mathcal{E}$ (written: $L \in \mathbf{NegO}^b\mathbf{TxtBc}^a(\mathbf{M})$) $\Leftrightarrow$ $(\exists S \subseteq \overline{L} \mid \text{card}(S) \leq b)(\forall I \mid \text{PosInfo}(I) = L \ \& \ S \subseteq \text{NegInfo}(I) \subseteq \overline{L})(\overset{\infty}{\forall} n)[W_{\mathbf{M}(I[n])} =^a L]$.

(b) $\mathbf{NegO}^b\mathbf{TxtBc}^a = \{\mathcal{L} \subseteq \mathcal{E} \mid (\exists \mathbf{M})[\mathcal{L} \subseteq \mathbf{NegO}^b\mathbf{TxtBc}^a(\mathbf{M})]\}$.

Clearly, for all a, $\mathbf{NegO}^0\mathbf{TxtEx}^a = \mathbf{TxtEx}^a$ and $\mathbf{NegO}^0\mathbf{TxtBc}^a = \mathbf{TxtBc}^a$.

Theorem 5.1 below shows that the $\mathbf{NegO}^*$ criteria are equivalent to supplying *all* the negative (as well as the positive) information to a learning machine.

Theorem 5.1. *(Baliga, Case and Jain[1]) For all* $a \in N \cup \{*\}$, $\mathbf{NegO}^*\mathbf{TxtEx}^a = \mathbf{InfEx}^a$ *and* $\mathbf{NegO}^*\mathbf{TxtBc}^a = \mathbf{InfBc}^a$.

Thus, in particular we have $\mathcal{E} \in \mathbf{NegO}^*\mathbf{TxtBc}^*$, and $\mathbf{NegO}^*\mathbf{TxtEx} \subseteq \mathbf{NegF}^1\mathbf{TxtEx}$.

Note that if we consider languages such that informant for a language can be effectively obtained from its text, then above theorem shows that $\mathbf{NegO}$ type negative data does not help.

As a corollary to Theorem 5.1, using Theorems 2.1 and 3.3, we have

Corollary 5.1. *(Baliga, Case and Jain[1])*

(a) For all $n \in N$, $\mathbf{TxtEx}^{n+1} - \mathbf{NegO}^*\mathbf{TxtEx}^n \neq \emptyset$;

(b) For all $n \in N$, $\mathbf{TxtBc}^{n+1} - \mathbf{NegO}^*\mathbf{TxtBc}^n \neq \emptyset$;

(c) $\mathbf{TxtBc} - \mathbf{NegO}^*\mathbf{TxtEx}^* \neq \emptyset$;

(d) $\mathbf{NegF}^1\mathbf{TxtEx} - \mathbf{NegO}^*\mathbf{TxtBc}^n \neq \emptyset$;

(e) $\mathbf{NegF}^1\mathbf{TxtEx} - \mathbf{NegO}^*\mathbf{TxtEx}^* \neq \emptyset$.

The above Corollary shows that there are classes of languages which can be learned with $n+1$ mistakes, but not with n, no matter how much open

negative information is provided in the n mistake case. In other words, the gap left by the possible extra anomaly can be greater in information content than the information provided by open negative information.

The following theorem generalizes Theorem 2.1(f).

Theorem 5.2. *(Baliga, Case and Jain[1]) For all $a \in N \cup \{*\}$ and $j \in N$, $[\mathbf{NegO}^a\mathbf{TxtEx}^{2j} \subset \mathbf{NegO}^a\mathbf{TxtBc}^j]$.*

The following result contrasts with Theorem 2.1(g).

Theorem 5.3. *(Baliga, Case and Jain[1]) $\mathbf{TxtEx}^* \subset \mathbf{NegO}^1\mathbf{TxtBc}$.*

The next theorem contrasts nicely with Theorem 5.1 above. It provides classes of languages which can be learned with $n+1$ pieces of core open negative information, but not with n, no matter how many anomalies are permitted in the n piece case. In other words, the extra possible negative information can be greater in information content than the information that may be omitted by the anomalies.

Theorem 5.4. *(Baliga, Case and Jain[1])*
(a) $\mathbf{NegO}^1\mathbf{TxtEx} - \mathbf{NegO}^0\mathbf{TxtBc}^* \neq \emptyset$.
(b) For all $n \in N$, $\mathbf{NegO}^{n+1}\mathbf{TxtEx} - \mathbf{NegO}^n\mathbf{TxtEx}^* \neq \emptyset$.
(c) For all $n \in N$, $\mathbf{NegO}^{n+1}\mathbf{TxtEx} - \bigcup_{j\in N} \mathbf{NegO}^n\mathbf{TxtBc}^j \neq \emptyset$.

The previous theorem has the following straightforward corollary.

Corollary 5.2. *(Baliga, Case and Jain[1]) For all $a \in N \cup \{*\}$ and $j, n \in N$,*
(a) $\mathbf{NegO}^n\mathbf{TxtEx}^a \subset \mathbf{NegO}^{n+1}\mathbf{TxtEx}^a$ and
(b) $\mathbf{NegO}^n\mathbf{TxtBc}^j \subset \mathbf{NegO}^{n+1}\mathbf{TxtBc}^j$.

5.1. *Complexity Advantages of Open Negative Information*

McNeill[17] posits that there is faster learning of language for children in homes in which more corrections (usually in the form of, possibly exemplary, expansions) are given. These corrections are, in part, a form of negative information. Theorem 5.5 below shows that an improvement in *speed* (measured by mind-changes) can result from the presence of open negative information even when the classes themselves can be learned without the negative information.

For this section it is convenient to modify the definition of the learning machine to the following.

Definition 5.3. A *language learning machine* is an algorithmic device which computes a mapping from SEQ (or SEG) into $N \cup \{?\}$.

Intuitively the outputted ?s represent the machine not yet committing to an output. This avoids biasing the number of mind changes before a learning machine converges.

In the next definition, the subscript b represents a bound on the number of mind changes allowed before convergence.

Definition 5.4. (Case and Smith[7], Case and Lynes[6]) Suppose $a, b \in N \cup \{*\}$. We say that $\mathbf{M}$ $\mathbf{TxtEx}^a_b$*-identifies* $L \Leftrightarrow [[L \in \mathbf{TxtEx}^a(\mathbf{M})] \wedge (\forall \text{ texts } T \text{ for } L)[\text{card}(\{x \mid [? \neq \mathbf{M}(T[x])] \wedge [\mathbf{M}(T[x]) \neq \mathbf{M}(T[x+1])]\}) \leq b]]$.

One can similarly define $\mathbf{NegO}^c\mathbf{TxtEx}^a_b$.

Next theorem shows the speed advantage of having open negative information.

Theorem 5.5. *(Baliga, Case and Jain[1]) There exists a class of languages $\mathcal{L}$ such that,*

(a) $\mathcal{L} \in \mathbf{TxtEx}$,

(b) $\mathcal{L} \in \mathbf{NegO}^1\mathbf{TxtEx}_0$, *and*

(c) $\mathcal{L} \notin \bigcup_{n \in N} \mathbf{TxtEx}^*_n$.

We now list some of the open problems regarding this model.

(a) For $i \geq 1$, $\mathcal{E} \in \mathbf{NegO}^i\mathbf{TxtBc}^*$? Here note that $\mathcal{E} \in \mathbf{NegO}^*\mathbf{TxtBc}^*$.

(b) By Theorem 2.1(g), $\mathbf{TxtEx}^{2j+1} - \mathbf{TxtBc}^j \neq \emptyset$. Similarly, can it be shown that, for $i \geq 1$, $\mathbf{NegO}^i\mathbf{TxtEx}^{2j+1} - \mathbf{NegO}^i\mathbf{TxtBc}^j \neq \emptyset$?

(c) For $i \geq 1$, is $\mathbf{NegO}^i\mathbf{TxtEx}^* \subset \mathbf{NegO}^{i+1}\mathbf{TxtBc}$? So far we know that $\mathbf{NegO}^*\mathbf{TxtEx}^* \subset \mathbf{NegO}^*\mathbf{TxtBc}$.

6. Learning with Negative Counterexamples

We now consider providing negative data to the learner via counterexamples to the conjectures of the learner. We will be considering three variants of the model. Intuitively, for learning with negative counterexamples, 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. The list of negative counterexamples may be modeled as a second text 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, $\mathbf{M}(T[n], T'[n]) = i$.

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

Definition 6.1. (Jain and Kinber[13]) Suppose $a \in N \cup \{*\}$.

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

We also consider two variants of above definition as follows:

— the learner gets least negative counterexample instead of any counterexample. This criteria is denoted $\mathbf{LNCEx}^a$.

— the negative counterexample is provided only if there exists one such counterexample $\leq$ the maximum positive element seen in the input so far (otherwise the learner gets #). This criteria is denoted by $\mathbf{BNCEx}^a$. (Essentially S_n in the definition of $T'(n)$ in part (a) is replaced by $S_n = \overline{L} \cap W_{\mathbf{M}(T[n],T'[n])} \cap \{x \mid x < \max(\text{content}(T[n]))\}$). The **BNC** model essentially addresses some complexity constraints.

Similarly, we can define $\mathbf{NCBc}^a$, $\mathbf{LNCBc}^a$ and $\mathbf{BNCBc}^a$ criteria of inference.

It is easy to see that $\mathbf{TxtEx}^a \subseteq \mathbf{BNCEx}^a \subseteq \mathbf{NCEx}^a \subseteq \mathbf{LNCEx}^a$. All of these containments, except the last one, are proper.

Part (a) of the following theorem shows that every indexed family can be learned using positive data and negative counterexamples. This improves a classical result that every indexed family is learnable from informants. Since there exist indexed families not in **TxtEx**, this illustrates a difference between **NCEx** learning and learning without negative counterexamples.

Part (b) of the following theorem illustrates another difference between **NCEx** learning and **TxtEx** learning. Such a result does not hold for **TxtEx** (for example, $\{F \mid F \text{ is finite }\} \cup \{L\} \notin \mathbf{TxtEx}$, for any infinite language L).

Theorem 6.1. *(Jain and Kinber[13])*

(a) Suppose $\mathcal{L}$ is an indexed family. Then $\mathcal{L} \in \mathbf{NCEx}$.

(b) Suppose $\mathcal{L} \in \mathbf{NCEx}$ *and* L *is a recursive language. Then* $\mathcal{L} \cup \{L\} \in \mathbf{NCEx}$.

Part (b) of the above theorem does not generalize to taking r.e. language (instead of recursive language) L, as witnessed by $\mathcal{L} = \{\{A \cup \{x\}\} \mid x \notin A\}$, and $L = A$, where A is any non-recursive r.e. set. Here note that $\mathcal{L} \in \mathbf{TxtEx}$, but $\mathcal{L} \cup \{L\}$ is not in $\mathbf{NCEx}$.

The following theorem shows that using least negative counterexamples, rather than arbitrary negative counterexamples, does not enhance power of a learner.

Theorem 6.2. *(Jain and Kinber*[13]*) Let* $a \in N \cup \{*\}$. *Then,* $\mathbf{NCEx}^a = \mathbf{LNCEx}^a \subseteq \mathbf{InfEx}^a$.

For **Bc**-style learning, a limited version of above holds. Though, the equality $\mathbf{NCBc} = \mathbf{LNCBc}$ can be generalized to learning with anomalies (see Corollary 6.2 below), $\mathbf{LNCBc} \subseteq \mathbf{InfBc}$, cannot be generalized to learning with anomalies.

Proposition 6.1. *(Jain and Kinber*[13]*)* $\mathbf{NCBc} = \mathbf{LNCBc} \subseteq \mathbf{InfBc}$.

Part (a) of the following theorem shows that all classes of languages learnable in the basic **Ex**-style model with arbitrary finite number of errors in almost all conjectures can be learned without errors in the basic **Bc**-style model. This contrasts with learning from texts where $\mathbf{TxtEx}^{2j+1} - \mathbf{TxtBc}^j \neq \emptyset$ (Theorem 2.1(g)).

Part (b) of the following theorem is somewhat surprising. It shows that sometimes negative counterexamples are not enough: to learn a language, the learner must have access to *all* negative examples.

Theorem 6.3. *(Jain and Kinber*[13]*)*
(a) $\mathbf{NCEx}^* \subseteq \mathbf{NCBc}$.
(b) $\mathbf{InfEx} - \mathbf{NCBc} \neq \emptyset$.

We now show advantages of having negative counterexamples. Part (a) of the following theorem shows that the model **BNCEx** is quite powerful: there are classes of languages learnable in this model that cannot be learned in the classical **Bc**-style model even when an arbitrary finite number of errors is allowed in almost all conjectures. Part (b) of the following theorem shows that there are classes of languages learnable in the basic model that cannot be learned in any of the models that use negative counterexamples of limited size.

Theorem 6.4. *(Jain and Kinber[13])*
(a) $\mathbf{BNCEx} - \mathbf{TxtBc}^* \neq \emptyset$.
(b) $\mathbf{NCEx} - \mathbf{BNCBc}^* \neq \emptyset$.

Note that the diagonalizations in Theorem 6.4 can be shown using indexed families of languages. Thus, in contrast to Theorem 6.1, there exists an indexed family not in $\mathbf{BNCBc}^*$.

In contrast to Theorem 6.4 (b), the following shows that if attention is restricted to only infinite languages, then **NCEx** and **BNCEx** behave similarly.

Theorem 6.5. *(Jain and Kinber[13]) Suppose* $a \in N \cup \{*\}$. *Suppose* $\mathcal{L}$ *consists of only infinite languages. Then*
(a) $\mathcal{L} \in \mathbf{NCEx}^a$ *iff* $\mathcal{L} \in \mathbf{BNCEx}^a$.
(b) $\mathcal{L} \in \mathbf{NCBc}^a$ *iff* $\mathcal{L} \in \mathbf{BNCBc}^a$.

We now consider the error hierarchy for learning with negative counterexamples. That is, learning with at most $n+1$ errors in almost all conjectures in the basic model is stronger than learning with at most n errors. The hierarchy easily follows from the following theorem.

Theorem 6.6. *(Jain and Kinber[13]) Suppose* $n \in N$.
(a) $\mathbf{TxtEx}^{n+1} - \mathbf{NCEx}^n \neq \emptyset$.
(b) $\mathbf{TxtEx}^* - \bigcup_{n \in N} \mathbf{NCEx}^n \neq \emptyset$.
(c) $\mathbf{TxtBc} - \mathbf{NCEx}^* \neq \emptyset$.
(d) $\mathbf{TxtBc}^1 - \mathbf{NCBc} \neq \emptyset$.

As, $\mathbf{TxtEx}^{n+1} \subseteq \mathbf{BNCEx}^{n+1} \subseteq \mathbf{NCEx}^{n+1} \subseteq \mathbf{LNCEx}^{n+1}$, the following corollary follows from Theorem 6.6.

Corollary 6.1. *(Jain and Kinber[13]) Suppose* $n \in N$. *Then, for* $\mathbf{I} \in \{\mathbf{NCEx}, \mathbf{LNCEx}, \mathbf{BNCEx}\}$, *we have* $\mathbf{I}^n \subset \mathbf{I}^{n+1}$.

Now we consider another surprising result. There exists a $\mathbf{Bc}^1$-style learner with negative counterexamples, with the "ultimate power" - it can learn the class of all recursively enumerable languages!

Theorem 6.7. *(Jain and Kinber[13])* $\mathcal{E} \in \mathbf{NCBc}^1$.

Since $\mathcal{E} \in \mathbf{InfBc}^*$, we have

Corollary 6.2. *(Jain and Kinber[13]) (a)* $\mathbf{NCBc}^1 = \mathbf{InfBc}^*$.
(b) For all $a \in N \cup \{*\}$, $\mathbf{NCBc}^a = \mathbf{LNCBc}^a$.

The following corollary shows a contrast with respect to the case when there are no errors in conjectures (Proposition 6.1 and Theorem 6.3(c)). What a difference just one error can make!

Corollary 6.3. *(Jain and Kinber[13]) For all $n \in N$, $n > 0$,* $\mathbf{InfBc}^n \subset \mathbf{NCBc}^n = \mathbf{NCBc}^1$.

Based on the ideas similar to the ones used for proving Theorem 6.7, one can show

Theorem 6.8. *(Jain and Kinber[13]) (a) Let $\mathcal{L} = \{L \in \mathcal{E} \mid L \text{ is infinite }\}$. Then* $\mathcal{L} \in \mathbf{BNCBc}^1$.
(b) For all $n \in N$, $\mathbf{TxtBc}^n \subseteq \mathbf{BNCBc}^1$.
(c) $\mathbf{TxtEx}^* \subseteq \mathbf{BNCBc}^1$.

As there exists a class of infinite languages which does not belong to $\mathbf{InfBc}^n$ (see Case and Smith[7]), we have

Corollary 6.4. *(Jain and Kinber[13]) For all $n \in N$,* $\mathbf{BNCBc}^1 - \mathbf{InfBc}^n \neq \emptyset$.

Thus, $\mathbf{BNCBc}^m$ and $\mathbf{InfBc}^n$ are incomparable for $m > 0$, $m, n \in N$. The above result does not generalize to $\mathbf{InfBc}^*$, as $\mathbf{InfBc}^*$ contains the class $\mathcal{E}$.

We now mention some of the open questions regarding behaviourally correct learning when the size of the negative counterexamples is bounded.
(a) Is $\mathbf{BNCBc}^n$ hierarchy strict?
(b) Is $\mathbf{TxtBc}^* \subseteq \mathbf{BNCBc}^1$?

6.1. *Complexity Issues*

We now consider the complexity advantages of having negative counterexamples. This section is based on the paper[13].

The class $\mathcal{L}_1 = \{L \mid \text{card}(N - L) = 1\}$ is in **TxtEx**, but requires unbounded number of mind changes to learn. On the other hand, $\mathcal{L}_1$ can be easily learned using one mind change if negative counterexamples are available. Thus, not only does **NCEx** model give learnability advantages over **TxtEx**, it also gives complexity advantages over **TxtEx** for some classes in **TxtEx**. Note that if one does not allow mind changes, then **NCEx** and **TxtEx** are both the same — thus the above result is the best mind change complexity advantage possible.

The class $\mathcal{L}_2 = \{L \mid (\exists i)[L = \{x \mid x \leq i\}]\} \cup \{N\}$, is learnable in **NCEx** model, but the number of mind changes is unbounded. However, $\mathcal{L}_2$ can be learned by using at most one mind change in the model **LNCEx**. Thus, even though **LNCEx** does not give learnability advantages over **NCEx**, it does give complexity advantages.

Let $a \dot{-} b = a - b$, if $a \geq b$; $a \dot{-} b = 0$ otherwise; Consider the class: $\mathcal{L}_3 = \{L \mid (\exists! e)[\langle 0, e\rangle \in L \ \wedge \ L - \{\langle 0, e\rangle\} \subseteq \{\langle x, y\rangle \mid x > 1\} \ \wedge \ \text{card}(L - \{\langle 0, e\rangle\}) = e \dot{-} \min(W_e)]\}$

$\mathcal{L}_3$ is in **LNCEx** with at most one mind change. However $\mathcal{L}_3$ cannot be learned in **InfEx** using bounded number of mind changes. Note that **LNCEx** $\subset$ **InfEx**. So getting negative counterexamples gives complexity advantages over informants, despite informant being more advantageous for learning as a whole.

The situation is more complex in considering the complexity advantages of **NCEx**-model compared to **InfEx** model. There exist classes which can be **NCEx**-identifies using $n-1$ mind changes, but cannot be **InfEx**-identified using $(2^n-1)-2$ mind changes. This is optimal as it can be shown that any class which can be **NCEx**-identified using $n-1$ mind changes can also be identified using $(2^n-1)-1$ mind changes in **InfEx**-model. We omit the details.

7. Learning With Subset Queries

We now consider learning with subset queries, which turn out to be another mechanism for providing negative examples. In this model learner is allowed to ask queries of the form "is $Q \subseteq L$?", where L is the language being learned.

If the answer to query is "no", we additionally can have the following possibilities:

(a) Learner is given an arbitrary counterexample (a member of $Q - L$);

(b) Learner is given the least counterexample;

(c) Learner is just given the answer 'no', without any counterexample.

We would often also consider bounds on the number of queries. We first formalize the definition of a learner which uses queries.

Definition 7.1. (Jain and Kinber[12]) A learner using queries can ask a query of form "$W_j \subseteq L$?" on any input σ. Answer to the query is "yes" or "no" (along with a possible counterexample). Then, based on input σ and answers received for queries made on prefixes of σ, **M** outputs a conjecture (from N).

Note that the queries are for recursively enumerable languages, which are posed to the teacher using a grammar (index) for the language. Many of the diagonalization results stand even if one uses arbitrary type of query language. However simulation results often crucially depend on the queries being made only via grammars for the queried languages.

Here, if one allows infinite number of subset queries, then one can learn the whole class $\mathcal{E}$ of recursively enumerable languages in **Ex**-model of learning. Furthermore, as we will see below (Proposition 7.2) if one allows finite, but unbounded, number of queries, then for **Ex**-model of learning the notion coincides with learning from negative counterexamples.

We now formalize learning via subset queries.

Definition 7.2. (Jain and Kinber[12]) Suppose $a \in N \cup \{*\}$.

(a) $\mathbf{M}$ $\mathbf{SubQ}^a\mathbf{Ex}$-*identifies* a language L (written: $L \in \mathbf{SubQ}^a\mathbf{Ex}(\mathbf{M})$) iff for any text T for L, it behaves as follows:

> (i) The number of queries $\mathbf{M}$ asks on prefixes of T is bounded by a (if $a = *$, then the number of such queries is finite). Furthermore, all the queries are of the form "$W_j \subseteq L$?"
>
> (ii) Suppose the answers to the queries are made as follows. For a query "$W_j \subseteq L$?", the answer is "yes" if $W_j \subseteq L$, and the answer is "no" if $W_j - L \neq \emptyset$. For "no" answers, $\mathbf{M}$ is also provided with a counterexample, $x \in W_j - L$. Then, for some k such that $W_k = L$, for all but finitely many n, $\mathbf{M}(T[n])$ outputs the grammar k.

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

$\mathbf{LSubQ}^a\mathbf{Ex}$-identification and $\mathbf{ResSubQ}^a\mathbf{Ex}$-identification can be defined similarly, where for $\mathbf{LSubQ}^a\mathbf{Ex}$-identification the learner gets the least counterexample for "no" answers, and for $\mathbf{ResSubQ}^a\mathbf{Ex}$-identification, the learner does not get any counterexample along with the "no" answers.

For $a, b \in N \cup \{*\}$, for $\mathbf{I} \in \{\mathbf{Ex}^b, \mathbf{Bc}^b\}$, one can similarly define $\mathbf{SubQ}^a\mathbf{I}$, $\mathbf{LSubQ}^a\mathbf{I}$, and $\mathbf{ResSubQ}^a\mathbf{I}$.

Next two propositions show a close correspondence between learning via negative counterexamples and learning via subset queries. In particular, learning via finite number of subset queries coincides with learning via negative counterexamples for **Ex**-model of learning.

Proposition 7.1. *(Jain and Kinber[12]) For any* $a \in N \cup \{*\}, \mathbf{I} \in \{\mathbf{Ex}^a, \mathbf{Bc}^a\}$,

(a) $\mathbf{SubQ}^*\mathbf{I} \subseteq \mathbf{NCI}$.

(b) $\mathbf{LSubQ}^*\mathbf{I} \subseteq \mathbf{LNCI}$.

(c) $\mathbf{ResSubQ}^*\mathbf{I} \subseteq \mathbf{ResNCI}$.

Proposition 7.2. *(Jain and Kinber[12]) Suppose* $a \in N \cup \{*\}$. $\mathbf{NCEx}^a = \mathbf{SubQ}^*\mathbf{Ex}^a = \mathbf{LNCEx}^a = \mathbf{LSubQ}^*\mathbf{Ex}^a = \mathbf{ResNCEx}^a = \mathbf{ResSubQ}^*\mathbf{Ex}^a$.

Next theorem establishes a hierarchy of learning capabilities with respect to the number of subset queries.

Theorem 7.1. *(Jain and Kinber[12]) Suppose* $n \in N$. *Then,* $\mathbf{ResSubQ}^{n+1}\mathbf{Ex} - \mathbf{LSubQ}^n\mathbf{Bc}^* \neq \emptyset$.

We now consider relationship between various types of subset queries. When only a single query or an unbounded but finite number of queries are used, different types of counterexamples do not make a difference.

Theorem 7.2. *(Jain and Kinber[12]) Suppose* $a \in N \cup \{*\}$, $b \in \{0, 1, *\}$, *and* $\mathbf{I} \in \{\mathbf{Ex}^a, \mathbf{Bc}^a\}$. *Then,* $\mathbf{ResSubQ}^b\mathbf{I} = \mathbf{SubQ}^b\mathbf{I} = \mathbf{LSubQ}^b\mathbf{I}$.

Thus, one needs to consider at least two queries when showing differences between various types of subset queries. The following theorem establishes the relationship between different types of subset queries.

Theorem 7.3. *(Jain and Kinber[12]) For all* $n \in N$,

(a) $\mathbf{LSubQ}^2\mathbf{Ex} - \mathbf{SubQ}^n\mathbf{Bc}^* \neq \emptyset$.

(b) $\mathbf{SubQ}^2\mathbf{Ex} - \mathbf{ResSubQ}^n\mathbf{Bc}^* \neq \emptyset$.

We next consider the anomaly hierarchy for the subset query learning criteria.

Theorem 7.4. *(Jain and Kinber[12]) (a) For all* $n \in N$, $\mathbf{TxtEx}^{n+1} - \mathbf{LSubQ}^*\mathbf{Ex}^n \neq \emptyset$.

(b) For all $n \in N$, $\mathbf{TxtBc}^{n+1} - \mathbf{LSubQ}^*\mathbf{Bc}^n \neq \emptyset$.

(c) $\mathbf{LSubQ}^*\mathbf{Ex}^* \subseteq \mathbf{ResSubQ}^*\mathbf{Bc}$.

As a corollary we get:

Corollary 7.1. *(Jain and Kinber[12]) Let* $a \in N \cup \{*\}$, *and* $n \in N$.

(a) $\mathbf{SubQ}^a\mathbf{Ex}^n \subset \mathbf{SubQ}^a\mathbf{Ex}^{n+1}$.

(b) $\mathbf{LSubQ}^a\mathbf{Ex}^n \subset \mathbf{LSubQ}^a\mathbf{Ex}^{n+1}$.
(c) $\mathbf{ResSubQ}^a\mathbf{Ex}^n \subset \mathbf{ResSubQ}^a\mathbf{Ex}^{n+1}$.

Similar corollary exists for **Bc**-criteria of learning with **Ex** being replaced by **Bc** in the above.

8. Random Negative Examples

In this section we briefly consider the impact of having random negative examples. It would be interesting to explore in general how random negative examples effect learning compared to other kind of negative examples as discussed in this paper.

When considering giving random negative examples, one may consider any measure theoretic method of selecting a random negative example. The only property used in the following is that if A is infinite and B is a finite subset of A, then measure of $A - B$ (with respect to A) is 1. Let $\mathbf{Rand}_p^1\mathbf{TxtEx}$ denote the class of languages that can be identified using positive data and one random negative example with probability p.

Theorem 8.1. *Consider the following class of languages:* $\mathcal{L} = \{L \mid (\exists i)[W_i = L \ \ \& \ \ \mathrm{card}(\overline{L} - \{\langle i, x\rangle \mid x \in N\}) < \infty \ \ \& \ \ \mathrm{card}(\overline{L} \cap \{\langle i, x\rangle \mid x \in N\}) = \infty]\}$. *Then,* $\mathcal{L} \in \mathbf{Rand}_1^1\mathbf{TxtEx} - \mathbf{TxtEx}$.

Note here that by Theorem 4.1, if one considers having arbitrary counterexamples, then for any class of languages which consists only of coinfinite languages, k arbitrary negative examples do not help in learning. So above theorem also shows that random negative examples are more useful for learning compared to arbitrary negative examples.

9. Acknowledgements

Sanjay Jain was supported in part by NUS grant number R252-000-127-112.

References

1. G. Baliga, J. Case, and S. Jain. Language learning with some negative information. *Journal of Computer and System Sciences*, 51(5):273–285, 1995.
2. 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.
3. L. Blum and M. Blum. Toward a mathematical theory of inductive inference. *Information and Control*, 28:125–155, 1975.

4. M. Blum. A machine-independent theory of the complexity of recursive functions. *Journal of the ACM*, 14:322–336, 1967.
5. R. Brown and C. Hanlon. Derivational complexity and the order of acquisition in child speech. In J. R. Hayes, editor, *Cognition and the Development of Language*. Wiley, 1970.
6. 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.
7. J. Case and C. Smith. Comparison of identification criteria for machine inductive inference. *Theoretical Computer Science*, 25:193–220, 1983.
8. M. Demetras, K. Post, and C. Snow. Feedback to first language learners: The role of repetitions and clarification questions. *Journal of Child Language*, 13:275–292, 1986.
9. M. Fulk. *A Study of Inductive Inference Machines*. PhD thesis, SUNY/Buffalo, 1985.
10. E. M. Gold. Language identification in the limit. *Information and Control*, 10:447–474, 1967.
11. K. Hirsh-Pasek, R. Treiman, and M. Schneiderman. Brown and Hanlon revisited: Mothers' sensitivity to ungrammatical forms. *Journal of Child Language*, 11:81–88, 1984.
12. S. Jain and E. Kinber. Learning languages from positive data and a finite number of queries. In Kamal Lodaya and Meena Mahajan, editors, *Foundations of Software Technology and Theoretical Computer Science*, volume 3328 of *Lecture Notes in Computer Science*, pages 360–372. Springer-Verlag, 2004.
13. S. Jain and E. Kinber. Learning languages from positive data and negative counterexamples. *Journal of Computer and System Sciences*, 2005. To appear.
14. S. Jain, D. Osherson, J. Royer, and A. Sharma. *Systems that Learn: An Introduction to Learning Theory*. MIT Press, Cambridge, Mass., second edition, 1999.
15. S. Jain and A. Sharma. Learning in the presence of partial explanations. *Information and Computation*, 95:162–191, 1991.
16. M. Machtey and P. Young. *An Introduction to the General Theory of Algorithms*. North Holland, New York, 1978.
17. D. McNeill. Developmental psycholinguistics. In F. Smith and G. Miller, editors, *The Genesis of Language*, pages 15–84. MIT Press, 1966.
18. T. Motoki. Inductive inference from all positive and some negative data. *Information Processing Letters*, 39(4):177–182, 1991.
19. D. Osherson, M. Stob, and S. Weinstein. Ideal learning machines. *Cognitive Science*, 6:277–290, 1982.
20. D. Osherson, M. Stob, and S. Weinstein. Learning theory and natural language. *Cognition*, 17:1–28, 1984.
21. D. Osherson, M. Stob, and S. Weinstein. *Systems that Learn: An Introduction*

to Learning Theory for Cognitive and Computer Scientists. MIT Press, 1986.
22. D. Osherson and S. Weinstein. Criteria of language learning. *Information and Control*, 52:123–138, 1982.
23. D. Osherson and S. Weinstein. A note on formal learning theory. *Cognition*, 11:77–88, 1982.
24. S. Pinker. Formal models of language learning. *Cognition*, 7:217–283, 1979.
25. H. Rogers. *Theory of Recursive Functions and Effective Computability*. McGraw-Hill, 1967. Reprinted, MIT Press 1987.
26. T. Shinohara. *Studies on Inductive Inference from Positive Data*. PhD thesis, Kyushu University, Kyushu, Japan, 1986.
27. K. Wexler. On extensional learnability. *Cognition*, 11:89–95, 1982.
28. K. Wexler and P. Culicover. *Formal Principles of Language Acquisition*. MIT Press, 1980.