Learning with Refutation

Sanjay Jain
Department of ISCS
National University of Singapore
Singapore 119260
Email: sanjay@iscs.nus.edu.sg

Abstract

In their pioneering work, Mukouchi and Arikawa modeled a learning situation in which the learner is expected to refute texts which are not representative of $\mathcal{L}$, the class of languages being identified. Lange and Watson extended this model to consider justified refutation in which the learner is expected to refute texts only if it contains a finite sample unrepresentative of the class $\mathcal{L}$. Both the above studies were in the context of indexed families of recursive languages. We extend this study in two directions. Firstly, we consider general classes of recursively enumerable languages. Secondly, we allow the machine to either identify or refute the unrepresentative texts (respectively, texts containing finite unrepresentative samples). We observe some surprising differences between our results and the results obtained for learning indexed families by Lange and Watson.

1 Introduction

Consider the identification of formal languages from positive data. A text for a language is a sequential presentation (in arbitrary oder) of all and only the elements of the language. In a widely studied identification paradigm, called **TxtEx**-identification, a learning machine is fed texts for languages, and, as the machine is receiving the data, it outputs a (possibly infinite) sequence of hypotheses. A learning machine is said to **TxtEx**-identify a language L just in case, when presented with a text for L, the sequence of hypotheses output by the machine converges to a grammar for L (formal definitions of criteria of inference informally presented in this section are given in Sections 2 and 3). A learning machine **TxtEx**-identifies a class, $\mathcal{L}$, of languages if it **TxtEx**-identifies each language in $\mathcal{L}$. This model of identification was introduced by Gold [Gol67] and has since then been explored by several researchers.

For the following, let $\mathcal{L}$ denote a class of languages which we want to identify. The model of identification presented above puts no constraint on the behaviour of the machine on texts for languages not in $\mathcal{L}$. However, we may want a machine to be able to detect that it cannot identify an input text for at least two reasons. Firstly, once a machine detects that it cannot identify an input text, we can use the machine for other useful purposes. Secondly, we may employ another machine to identify the input text, so as to further enhance the class of languages that can be identified. These are very useful considerations in the design of a practical learning system. Further, it is philosophically interesting to study machines which know their limitations.

In their pioneering work, Mukouchi and Arikawa [MA93] modeled such a scenario. They required that in addition to identifying all languages in $\mathcal{L}$, the machine should *refute* texts for languages not in $\mathcal{L}$ (i.e. texts which are "unrepresentative" of $\mathcal{L}$). We refer to this identification criterion as **TxtRef**. Mukouchi and Arikawa showed that **TxtRef** constitutes a serious drawback on the learning capabilities of machines. For example, a machine working as above cannot identify any infinite language.[1] This led Lange and Watson [LW94] (see also [MA95]) to consider justified refutation in which they require a machine to refute a text iff some initial segment of the text is enough to determine that the input text is not for a language in $\mathcal{L}$, i.e., the input text contains a finite sample "unrepresentative" of $\mathcal{L}$. We call this criteria of learning **TxtJRef**. Lange and Watson also considered a modification of justified refutation model (called **TxtJIRef**, for immediate justified refutation) in which the machine is required to refute the input text as soon as the initial segment contains an unrepresentative sample (formal definitions are given in Section 3). For further motivation regarding learning with refutation and its relationship with Popper's Logic for scientific inference, we refer the reader to [MA93] and [LW94].

[MA93] and [LW94] were mainly concerned with learning indexed families of recursive languages, where the hypothesis space is also an indexed family. In this paper, we extend the study in two directions. Firstly, we consider general classes of r.e. languages, and use the class of all computer programs (modeling accepting grammars) as the hypothesis space. Secondly, we allow a learning machine to either identify or refute unrepresentative texts (texts containing finite unrepresentative samples). Note that in the models of learning with refutation considered by [MA93] and [LW94] described above, the machine has to refute all texts which contain samples unrepresentative of $\mathcal{L}$. Thus, a machine which may identify some of these texts is disqualified.[2] For learning general classes of r.e. languages we feel that it is more reasonable to allow a machine to either identify or refute such texts (in most applications identifying an unrepresentative text is not going to be a disadvantage). This motivation has led us to the models described in the present paper. We refer to these criteria by attaching an **E** (for extended) in front of the corresponding criteria considered by [MA93, LW94].

We now highlight some important differences in the structure of results obtained by us, and those in [LW94]. In the context of learning indexed families of recursive languages, Lange and Watson (in their model, see also [MA95]) showed that **TxtJIRef** = **TxtJRef** (i.e. requiring machines to refute as soon as the initial segment becomes unrepresentative of $\mathcal{L}$, is not a restriction). Similar result was also shown by them for learning from informants[3]. We show that requiring immediate refutation is a restriction if we consider general classes of r.e. languages (in both our (extended) and Lange and Watson's models of justified refutation, and for learning from texts as well as informants). We also consider a variation of our model in which "unrepresentative" is with respect to what a machine identifies and not with respect to the class $\mathcal{L}$. In this variation, for learning from texts, (immediate) justified refutation model has the same power as **TxtEx** — a surprising result in the context of results in [LW94] and other results in this paper. However, in the context of learning from informants, even this variation fails to capture the power of **InfEx** (which is a criteria of learning from informants; see Section 2).

We now proceed formally.

[1] A machine working as above, cannot refute a text for any subset of a language it identifies; this along with a result due to Gold [Gol67] (which says that no machine can **TxtEx**-identify an infinite language and all of its finite subsets) shows that no machine can **TxtRef**-identify a class containing an infinite language.

[2] This property and the restriction to indexed families is crucially used in proving some of the results in [LW94].

[3] An informant for a language L is a sequential presentation of the elements of the set $\{(x,1) \mid x \in L\} \cup \{(x,0) \mid x \notin L\}$; see formal definition in Section 2.

2 Preliminaries

The recursion theoretic notions not explained below are from [Rog67]. $N = \{0, 1, 2, \ldots\}$ is the set of all natural numbers, and this paper considers r.e. subsets L of N. All conventions regarding range of variables apply, with or without decorations[4], unless otherwise specified. We let $c, e, i, j, k, l, m, n, p, s, t, u, v, w, x, y, z$, range over N. $\emptyset, \in, \subseteq, \supseteq, \subset, \supset$ denote empty set, member of, subset, superset, proper subset, and proper superset respectively. max(), min(), and card() denote the maximum, minimum, and cardinality of a set respectively, where by convention $\max(\emptyset) = 0$ and $\min(\emptyset) = \infty$. $\langle \cdot, \cdot \rangle$ stands for an arbitrary, one to one, computable encoding of all pairs of natural numbers onto N. Quantifiers $\forall^\infty, \exists^\infty$, and $\exists!$ denote for all but finitely many, there exist infinitely many, and there exists a unique respectively.

$\mathcal{R}$ denotes the set of total recursive functions from N to N. f and g range over total recursive functions. $\mathcal{E}$ denotes the set of all recursively enumerable (r.e.) sets. L ranges over $\mathcal{E}$. $\overline{L}$ denotes the complement of set L (i.e. $\overline{L} = N - L$). χ_L denotes the characteristic function of set L. $L_1 \mathbf{\Delta} L_2$ denotes the symmetric difference of L_1 and L_2, i.e., $L_1 \mathbf{\Delta} L_2 = (L_1 - L_2) \cup (L_2 - L_1)$. $\mathcal{L}$ ranges over subsets of $\mathcal{E}$. φ denotes a standard acceptable programming system (acceptable numbering) [Rog67]. φ_i denotes the function computed by the i-th program in the programming system φ. We also call i a program or index for φ_i. For a (partial) function η, domain(η) and range(η) respectively denote the domain and range of partial function η. We often write $\eta(x)\downarrow$ ($\eta(x)\uparrow$) to denote that $\eta(x)$ is defined (undefined). W_i denotes the domain of φ_i. W_i is considered as the language enumerated by the i-th program in φ system, and we say that i is a grammar or index for W_i. Φ denotes a standard Blum complexity measure [Blu67] for the programming system φ. $W_{i,s} = \{x < s \mid \Phi_i(x) < s\}$.

FIN denotes the class of finite languages, $\{L \mid \text{card}(L) < \infty\}$. **INIT** denotes the class of initial segments of N, that is $\{\{x \mid x < l\} \mid l \in N\}$. L is called a *single valued total language* iff $(\forall x)(\exists! y)[\langle x, y \rangle \in L]$. $\mathbf{svt} = \{L \mid L$ is a single valued total language $\}$. If $L \in \mathbf{svt}$, then we say that L represents the total function f such that $L = \{\langle x, f(x) \rangle \mid x \in N\}$. $\mathbf{K}$ denotes the set $\{x \mid \varphi_x(x)\downarrow\}$. Note that $\mathbf{K}$ is r.e. but $\overline{\mathbf{K}}$ is not.

A *text* is a mapping from N to $N \cup \{\#\}$. We let T range over texts. content(T) is defined to be the set of natural numbers in the range of T (i.e. content(T) = range(T) $- \{\#\}$). T is a *text for* L iff content(T) $= L$. That means a text for L is an infinite sequence whose range, except for a possible #, is just L.

An infinite *information sequence or informant* is a mapping from N to $(N \times \{0,1\}) \cup \{\#\}$. We let I range over informants. content(I) is defined to be the set of pairs in the range of I (i.e. content(I) = range(I) $- \{\#\}$). By PosInfo(I) we denote the set $\{x \mid (x, 1) \in \text{content}(I)\}$. By NegInfo($I$) we denote the set $\{x \mid (x, 0) \in \text{content}(I)\}$. For this paper, we only consider informants I such that PosInfo(I) and NegInfo(I) partition the set of natural numbers.

An *informant for* L is an informant I such that PosInfo(I) $= L$. It is useful to consider canonical information sequence for L. I is a canonical information sequence for L iff $I(x) = (x, \chi_L(x))$. We sometimes abuse notation and refer to the canonical information sequence for L by χ_L.

σ, τ, and γ range over finite initial segments of texts or informants, where the context determines which is meant. We denote the set of finite initial segments of texts by SEG and set of finite initial segments of informants by SEQ. We define content(σ) = range(σ) $- \{\#\}$ and, for $\sigma \in$ SEQ, PosInfo(σ) $= \{x \mid (x, 1) \in \text{content}(\sigma)\}$, and NegInfo($\sigma$) $= \{x \mid (x, 0) \in \text{content}(\sigma)\}$

[4]Decorations are subscripts, superscripts, primes and the like.

We use $\sigma \preceq T$ (respectively, $\sigma \preceq I$, $\sigma \preceq \tau$) to denote that σ is an initial segment of T (respectively, I, τ). $|\sigma|$ denotes the length of σ. $T[n]$ denotes the initial segment of T of length n. Similarly, $I[n]$ denotes the initial segment of I of length n. $\sigma \diamond \tau$ (respectively, $\sigma \diamond T$, $\sigma \diamond I$) denotes the concatenation of σ and τ (respectively, concatenation of σ and T, concatenation of σ and I). We sometimes abuse notation and say $\sigma \diamond w$ to denote the concatenation of σ with the sequence of one element w.

A *learning machine* (also called *inductive inference machine*) $\mathbf{M}$ is an algorithmic mapping from initial segments of texts (informants) to $(N \cup \{?\})$. We say that $\mathbf{M}$ converges on T to i, (written: $\mathbf{M}(T)\downarrow = i$) iff, for all but finitely many n, $\mathbf{M}(T[n]) = i$. Convergence on informants is defined similarly.

We now present the basic models of identification from texts and informants.

Definition 1 [Gol67, CL82]

(a) $\mathbf{M}$ $\mathbf{TxtEx}$-*identifies text* T iff $(\exists i \mid W_i = \text{content}(T))[\mathbf{M}(T)\downarrow = i]$.

(b) $\mathbf{M}$ $\mathbf{TxtEx}$-identifies L (written: $L \in \mathbf{TxtEx}(\mathbf{M})$) iff $\mathbf{M}$ $\mathbf{TxtEx}$-identifies each text T for L.

(c) $\mathbf{M}$ $\mathbf{TxtEx}$-identifies $\mathcal{L}$ iff $\mathbf{M}$ $\mathbf{TxtEx}$-identifies each $L \in \mathcal{L}$.

(d) $\mathbf{TxtEx} = \{\mathcal{L} \mid (\exists \mathbf{M})[\mathbf{M}\ \mathbf{TxtEx}\text{-identifies } \mathcal{L}]\}$.

Definition 2 [Gol67]

(a) $\mathbf{M}$ $\mathbf{TxtFin}$-*identifies text* T iff $(\exists i \mid W_i = \text{content}(T))(\exists n)[(\forall m < n)[\mathbf{M}(T[m]) = ?] \wedge (\forall m \geq n)[\mathbf{M}(T[m]) = i]]$.

(b) $\mathbf{M}$ $\mathbf{TxtFin}$-identifies L (written: $L \in \mathbf{TxtFin}(\mathbf{M})$) iff $\mathbf{M}$ $\mathbf{TxtFin}$-identifies each text T for L.

(c) $\mathbf{M}$ $\mathbf{TxtFin}$-identifies $\mathcal{L}$ iff $\mathbf{M}$ $\mathbf{TxtFin}$-identifies each $L \in \mathcal{L}$.

(d) $\mathbf{TxtFin} = \{\mathcal{L} \mid (\exists \mathbf{M})[\mathbf{M}\ \mathbf{TxtFin}\text{-identifies } \mathcal{L}]\}$.

Intuitively, for finite identification, $\mathbf{M}$ outputs just one grammar, which must be correct.

Definition 3 [Gol67, CL82]

(a) $\mathbf{M}$ $\mathbf{InfEx}$-*identifies informant* I iff $(\exists i \mid W_i = \text{PosInfo}(I))[\mathbf{M}(I)\downarrow = i]$.

(b) $\mathbf{M}$ $\mathbf{InfEx}$-identifies L (written: $L \in \mathbf{InfEx}(\mathbf{M})$) iff $\mathbf{M}$ $\mathbf{InfEx}$-identifies each informant I for L.

(c) $\mathbf{M}$ $\mathbf{InfEx}$-identifies $\mathcal{L}$ iff $\mathbf{M}$ $\mathbf{InfEx}$-identifies each $L \in \mathcal{L}$.

(d) $\mathbf{InfEx} = \{\mathcal{L} \mid (\exists \mathbf{M})[\mathbf{M}\ \mathbf{InfEx}\text{-identifies } \mathcal{L}]\}$.

Definition 4 [Gol67]

(a) $\mathbf{M}$ $\mathbf{InfFin}$-*identifies informant* I iff $(\exists i \mid W_i = \text{PosInfo}(I))(\exists n)[(\forall m < n)[\mathbf{M}(I[m]) = ?] \wedge (\forall m \geq n)[\mathbf{M}(I[m]) = i]]$.

(b) $\mathbf{M}$ $\mathbf{InfFin}$-identifies L (written: $L \in \mathbf{InfFin}(\mathbf{M})$) iff $\mathbf{M}$ $\mathbf{InfFin}$-identifies each informant I for L.

(c) $\mathbf{M}$ $\mathbf{InfFin}$-identifies $\mathcal{L}$ iff $\mathbf{M}$ $\mathbf{InfFin}$-identifies each $L \in \mathcal{L}$.

(d) $\mathbf{InfFin} = \{\mathcal{L} \mid (\exists \mathbf{M})[\mathbf{M}\ \mathbf{InfFin}\text{-identifies } \mathcal{L}]\}$.

The next two definitions introduce reliable identification. A reliable machine diverges on texts (informants) it does not identify. Though a reliable machine does not refute a text (informant) it does not identify, it at least doesn't give false hope by converging to a wrong hypothesis. This was probably the first constraint imposed on machine's behaviour on languages outside the class being identified. We give two variations of reliable identification based on whether the machine is expected to diverge on every text which is for a language not in $\mathcal{L}$, or just on texts it does not identify.

For the rest of the paper, for criteria of inference, $\mathbf{J}$, we will only define what it means for a machine to $\mathbf{J}$-identify a class of languages $\mathcal{L}$. The identification class $\mathbf{J}$ is then implicitly defined as $\mathbf{J} = \{\mathcal{L} \mid (\exists \mathbf{M})[\mathbf{M}\ \mathbf{J}\text{-identifies } \mathcal{L}]\}$.

Definition 5 [Min76]
(a) $\mathbf{M}$ **TxtRel**-identifies $\mathcal{L}$ iff
(a.1) $\mathbf{M}$ **TxtEx**-identifies $\mathcal{L}$ and
(a.2) $(\forall T \mid \text{content}(T) \notin \mathcal{L})[\mathbf{M}(T)\uparrow]$.
(b) $\mathbf{M}$ **InfRel** identifies $\mathcal{L}$ iff
(b.1) $\mathbf{M}$ **InfEx**-identifies $\mathcal{L}$ and
(b.2) $(\forall I \mid \text{PosInfo}(I) \notin \mathcal{L})[\mathbf{M}(I)\uparrow]$.
(c) $\mathbf{M}$ **ETxtRel**-identifies $\mathcal{L}$ iff
(c.1) $\mathbf{M}$ **TxtEx**-identifies $\mathcal{L}$ and
(c.2) $(\forall T \mid \mathbf{M}$ does not **TxtEx**-identify $T)[\mathbf{M}(T)\uparrow]$.
(d) $\mathbf{M}$ **EInfRel** identifies $\mathcal{L}$ iff
(d.1) $\mathbf{M}$ **InfEx**-identifies $\mathcal{L}$ and
(d.2) $(\forall I \mid \mathbf{M}$ does not **InfEx**-identify $I)[\mathbf{M}(I)\uparrow]$.

The following propositions are some known facts about the identification criteria discussed above, which we will be using in this paper. First two propositions are based on results due to Gold [Gol67].

Proposition 6 *Suppose L is any infinite r.e. language, and* $\mathbf{M}$ *a learning machine. Let σ be such that* $\text{content}(\sigma) \subseteq L$. *Then there exists an r.e. L',* $\text{content}(\sigma) \subseteq L' \subseteq L$ *such that* $\mathbf{M}$ *does not* **TxtEx**-*identify L'.*

Proposition 7 *Suppose L is any infinite r.e. language, and* $\mathbf{M}$ *a learning machine. Let σ be such that $PosInfo(\sigma) \subseteq L$. Then there exists an r.e. L', $PosInfo(\sigma) \subseteq L' \subseteq L$ such that* $\mathbf{M}$ *does not* **InfEx**-*identify L'.*

Proposition 8 *[Gol67, Sha98]* **TxtFin** $\subset$ **InfFin** $\subset$ **TxtEx** $\subset$ **InfEx**.

3 Learning with Refutation

In this section we introduce the refutation models for learning. For learning with refutation we allow learning machines to output a special refutation symbol denoted $\perp$. We assume that if $\mathbf{M}(\sigma) = \perp$, then, for all τ, $\mathbf{M}(\sigma \diamond \tau) = \perp$. Intuitively output of $\perp$ denotes that $\mathbf{M}$ is declaring the input to be "unrepresentative". In the following definitions we consider the different criteria mentioned in the introduction. It is useful to define $\mathbf{Cons}_{\mathcal{L}} = \{\sigma \mid (\exists L \in \mathcal{L})[\text{content}(\sigma) \subseteq L]\}$.

The following definition introduces learning with refutation for general classes of r.e. languages.

Definition 9 [MA93] $\mathbf{M}$ **TxtRef** identifies $\mathcal{L}$ iff
(a) $\mathbf{M}$ **TxtEx**-identifies $\mathcal{L}$ and
(b) $(\forall T \mid \text{content}(T) \notin \mathcal{L})[\mathbf{M}(T)\downarrow = \perp]$.

If $\mathbf{M}(T)\downarrow = \perp$, then we often say that $\mathbf{M}$ refutes the text T. The following definitions introduce identification with justified refutation for general classes of r.e. languages. Below **JRef** stands for justified refutation, and **JIRef** stands for justified immediate refutation.

Definition 10 [LW94] $\mathbf{M}$ **TxtJRef** identifies $\mathcal{L}$ iff
(a) $\mathbf{M}$ **TxtEx**-identifies $\mathcal{L}$ and
(b) $(\forall T \mid \text{content}(T) \notin \mathcal{L}$ and $(\exists \sigma \preceq T)[\sigma \notin \mathbf{Cons}_{\mathcal{L}}])[\mathbf{M}(T)\downarrow = \perp]$.

Intuitively, in the above definition, $\mathbf{M}$ is required to refute a text T only if T contains a finite sample which is unrepresentative of $\mathcal{L}$. Following definition additionally requires that $\mathbf{M}$ refutes an initial segment of T as soon as it contains an unrepresentative sample.

Definition 11 [LW94] $\mathbf{M}$ **TxtJIRef** identifies $\mathcal{L}$ iff
(a) $\mathbf{M}$ **TxtEx**-identifies $\mathcal{L}$ and
(b) $(\forall T \mid \text{content}(T) \notin \mathcal{L})(\forall \sigma \preceq T \mid \sigma \notin \mathbf{Cons}_{\mathcal{L}})[\mathbf{M}(\sigma) = \perp]$.

We now present the above criteria for learning from informants. It is useful to define the following analogue of **Cons**. $\mathbf{ICons}_{\mathcal{L}} = \{\sigma \mid (\exists L \in \mathcal{L})[\text{PosInfo}(\sigma) \subseteq L \ \wedge \ \text{NegInfo}(\sigma) \subseteq \overline{L}]\}$.

Definition 12
(a) [MA93] $\mathbf{M}$ **InfRef** identifies $\mathcal{L}$ iff
(a.1) $\mathbf{M}$ **InfEx**-identifies $\mathcal{L}$ and
(a.2) $(\forall I \mid \text{PosInfo}(I) \notin \mathcal{L})[\mathbf{M}(I)\downarrow = \perp]$.
(b) [LW94] $\mathbf{M}$ **InfJRef** identifies $\mathcal{L}$ iff
(b.1) $\mathbf{M}$ **InfEx**-identifies $\mathcal{L}$ and
(b.2) $(\forall I \mid \text{PosInfo}(I) \notin \mathcal{L}$ and $(\exists \sigma \preceq I)[\sigma \notin \mathbf{ICons}_{\mathcal{L}}])[\mathbf{M}(I)\downarrow = \perp]$.
(c) [LW94] $\mathbf{M}$ **InfJIRef** identifies $\mathcal{L}$ iff
(c.1) $\mathbf{M}$ **InfEx**-identifies $\mathcal{L}$ and
(c.2) $(\forall I \mid \text{PosInfo}(I) \notin \mathcal{L})(\forall \sigma \preceq I \mid \sigma \notin \mathbf{ICons}_{\mathcal{L}})[\mathbf{M}(\sigma) = \perp]$.

We now present our extended definition for learning with refutation. Intuitively, we extend the above definitions of [MA93, LW94] by allowing a machine to identify an unrepresentative text.

The following definition is a modification of the corresponding definition in [MA93]. **E** in the beginning of criteria of inference, such as **ETxtRef**, stands for extended.

Definition 13 $\mathbf{M}$ **ETxtRef** identifies $\mathcal{L}$ iff
(a) $\mathbf{M}$ **TxtEx**-identifies $\mathcal{L}$ and
(b) $(\forall T \mid \mathbf{M}$ does not **TxtEx**-identify $T)[\mathbf{M}(T)\downarrow = \perp]$.

Intuitively, in the above definition we require the machine to refute the input text, only if it does not **TxtEx**-identify it.

The following definitions on identification by justified refutation are modifications of corresponding definitions considered by [LW94].

Definition 14 $\mathbf{M}$ **ETxtJRef** identifies $\mathcal{L}$ iff
(a) $\mathbf{M}$ **TxtEx**-identifies $\mathcal{L}$ and
(b) $(\forall T \mid \mathbf{M}$ does not **TxtEx**-identify T and $(\exists \sigma \preceq T)[\sigma \notin \mathbf{Cons}_{\mathcal{L}}])[\mathbf{M}(T) = \perp]$.

Intuitively, in the above definition, $\mathbf{M}$ is required to refute a text T only if $\mathbf{M}$ does not identify T, and T contains a finite sample which is unrepresentative of $\mathcal{L}$. In the following definition, we additionally require that $\mathbf{M}$ refute an initial segment of T as soon as it contains an unrepresentative sample.

Definition 15 $\mathbf{M}$ **ETxtJIRef** identifies $\mathcal{L}$ iff
(a) $\mathbf{M}$ **TxtEx**-identifies $\mathcal{L}$ and
(b) $(\forall T \mid \mathbf{M}$ does not **TxtEx**-identify $T)(\forall \sigma \preceq T \mid \sigma \notin \mathbf{Cons}_{\mathcal{L}})[\mathbf{M}(\sigma) = \perp]$.

We now present the above criteria for learning from informants.

Definition 16
(a) $\mathbf{M}$ **EInfRef** identifies $\mathcal{L}$ iff
(a.1) $\mathbf{M}$ **InfEx**-identifies $\mathcal{L}$ and
(a.2) $(\forall I \mid \mathbf{M}$ does not **InfEx**-identify $I)[\mathbf{M}(I)\downarrow = \perp]$.
(b) $\mathbf{M}$ **EInfJRef** identifies $\mathcal{L}$ iff
(b.1) $\mathbf{M}$ **InfEx**-identifies $\mathcal{L}$ and
(b.2) $(\forall I \mid \mathbf{M}$ does not **InfEx**-identify I and $(\exists \sigma \preceq I)[\sigma \notin \mathbf{ICons}_{\mathcal{L}}])[\mathbf{M}(I)\downarrow = \perp]$.
(c) $\mathbf{M}$ **EInfJIRef** identifies $\mathcal{L}$ iff
(c.1) $\mathbf{M}$ **InfEx**-identifies $\mathcal{L}$ and
(c.2) $(\forall I \mid \mathbf{M}$ does not **InfEx**-identify $I)(\forall \sigma \preceq I \mid \sigma \notin \mathbf{ICons}_{\mathcal{L}})[\mathbf{M}(\sigma) = \perp]$.

4 Results

We next consider the relationship between different identification criteria defined in this paper. The results presented give a complete relationship between all the criteria of inference introduced in this paper.

4.1 Containment Results

The following containments follow immediately from the definitions.

Proposition 17 $\mathbf{TxtRef} \subseteq \mathbf{TxtRel} \subseteq \mathbf{TxtEx}$.
$\mathbf{TxtRef} \subseteq \mathbf{TxtJRef} \subseteq \mathbf{TxtEx}$.
$\mathbf{TxtJIRef} \subseteq \mathbf{TxtJRef} \subseteq \mathbf{TxtEx}$.
$\mathbf{InfRef} \subseteq \mathbf{InfRel} \subseteq \mathbf{InfEx}$.
$\mathbf{InfRef} \subseteq \mathbf{InfJRef} \subseteq \mathbf{InfEx}$.
$\mathbf{InfJIRef} \subseteq \mathbf{InfJRef} \subseteq \mathbf{InfEx}$.
$\mathbf{TxtRef} \subseteq \mathbf{InfRef} \subseteq \mathbf{InfEx}$.
$\mathbf{TxtRel} \subseteq \mathbf{InfRel} \subseteq \mathbf{InfEx}$.

Proposition 18 $\mathbf{ETxtRef} \subseteq \mathbf{ETxtRel} \subseteq \mathbf{TxtEx}$.
$\mathbf{ETxtRef} \subseteq \mathbf{ETxtJRef} \subseteq \mathbf{TxtEx}$.
$\mathbf{ETxtJIRef} \subseteq \mathbf{ETxtJRef} \subseteq \mathbf{TxtEx}$.
$\mathbf{EInfRef} \subseteq \mathbf{EInfRel} \subseteq \mathbf{InfEx}$.
$\mathbf{EInfRef} \subseteq \mathbf{EInfJRef} \subseteq \mathbf{InfEx}$.
$\mathbf{EInfJIRef} \subseteq \mathbf{EInfJRef} \subseteq \mathbf{InfEx}$.
$\mathbf{ETxtRef} \subseteq \mathbf{EInfRef} \subseteq \mathbf{InfEx}$.
$\mathbf{ETxtRel} \subseteq \mathbf{EInfRel} \subseteq \mathbf{InfEx}$.

Proposition 19 *(a)* **TxtJIRef** $=$ **ETxtJIRef***.*
(b) **InfJIRef** $=$ **EInfJIRef***.*

PROOF. (a) It suffices to show **ETxtJIRef** $\subseteq$ **TxtJIRef**. Suppose **M** **ETxtJIRef**-identifies $\mathcal{L}$.

Claim 20 *For all σ such that $\sigma \notin \mathbf{Cons}_{\mathcal{L}}$, $\mathbf{M}(\sigma) = \perp$.*

PROOF. (of Claim) Suppose by way of contradiction that there is a σ such that $\sigma \notin \mathbf{Cons}_{\mathcal{L}}$ and $\mathbf{M}(\sigma) \neq \perp$. Let T be an extension of σ such that **M** does not **TxtEx**-identify T. Note that by Proposition 6 there exists such a T. But then by definition of **ETxtJIRef**, $\mathbf{M}(\sigma) = \perp$. A contradiction. Thus claim holds. □

It immediately follows from the claim that **M** also **TxtJIRef**-identifies $\mathcal{L}$.

Part (b) can be proved in a manner similar to part (a). ∎

Theorem 21 *Suppose X is a set not in Σ_2. Let $\mathcal{L} = \{\{i\} \mid i \in X\}$. Suppose* $\mathbf{J} \in \{$ **TxtRef***,* **TxtRel***,* **TxtJRef***,* **InfRef***,* **InfRel***,* **InfJRef**$\}$*. Then $\mathcal{L} \in$* **EJ** *but $\mathcal{L} \notin$* **J***.*

PROOF. It is easy to construct a machine which identifies all texts for empty or singleton languages and refutes/diverges on all texts for languages containing at least 2 elements. Thus, we have that $\mathcal{L} \in \mathbf{EJ}$.

Now, suppose by way of contradiction that **M** **J**-identifies $\mathcal{L}$. Then, $i \in X$ iff $(\exists \sigma \mid \text{content}(\sigma) = \{i\})(\forall \tau \mid \sigma \preceq \tau \ \wedge \ \text{content}(\tau) = \{i\})[\mathbf{M}(\sigma) = \mathbf{M}(\tau) \ \wedge \ \mathbf{M}(\sigma) \in N]$. A contradiction to the fact that X is not in Σ_2. ∎

Theorem 22 *Suppose* $\mathbf{J} \in \{$ **TxtRef***,* **TxtRel***,* **TxtJRef***,* **InfRef***,* **InfRel***,* **InfJRef**$\}$*. Then,* $\mathbf{J} \subset \mathbf{EJ}$*.*

PROOF. $\mathbf{J} \subseteq \mathbf{EJ}$ follows immediately from definition. Proper containment follows from Theorem 21. ∎

4.2 Separation Results

We now proceed to show the separation results. The next two theorems show the advantages of finite identification over reliable identification and identification with refutation. For the proof of first theorem, we need the following proposition, which follows immediately from definitions.

Proposition 23 *Suppose $\mathcal{L}$ is such that:*
(a) $\mathcal{L} \in$ **EInfJRef***, and*
(b) For all $L_1, L_2 \in \mathcal{L}$, either $L_1 \cap L_2 = \emptyset$ or $L_1 = L_2$.
Then $\mathcal{L} \in$ **EInfRef** *(and thus in* **EInfRel***).*

Let $\mathbf{M}_0, \mathbf{M}_1, \ldots$, denote a recursive enumeration of all machines.

Theorem 24 **TxtFin** $-$ (**EInfRel** $\cup$ **EInfJRef**) $\neq \emptyset$*.*

PROOF. For each i, we will define below a nonempty language L_i with the following properties:

(a) $L_i \subseteq \{\langle i, n\rangle \mid n \in N\}$;

(b) either $\mathbf{M}_i$ is not reliable, or $L_i \notin \mathbf{InfEx}(\mathbf{M}_i)$.

(c) a grammar for L_i can be obtained effectively in i.

We take $\mathcal{L} = \{L_i \mid i \in N\}$. Clearly, $\mathcal{L} \in \mathbf{TxtFin}$ (since a grammar for L_i can be found effectively from i). Further, (using clause (b) above) we have that $\mathcal{L} \notin \mathbf{EInfRel}$. It thus follows from Proposition 23 that $\mathcal{L} \notin \mathbf{EInfJRef}$.

We will define L_i in stages below. Let L_i^s denote L_i defined before stage s. Let $L_i^0 = \{\langle i, 0\rangle\}$. Let $x^0 = \langle i, 1\rangle$. Go to stage 0.

Stage s

1. Suppose, I_1^s is the canonical information sequence for L_i^s and I_2^s is the canonical information sequence for $L_i^s \cup \{x^s\}$.
2. Search for $n > x^s$ such that either $\mathbf{M}_i(I_1^s[n]) \neq \mathbf{M}_i(I_1^s[x^s])$ or $\mathbf{M}_i(I_2^s[n]) \neq \mathbf{M}_i(I_2^s[x^s])$.

   If and when such an n is found proceed to step 3.
3. Let n be as found in step 2. If $\mathbf{M}_i(I_2^s[n]) \neq \mathbf{M}_i(I_2^s[x^s])$, then let $L_i^{s+1} = L_i^s \cup \{x^s\}$; otherwise let $L_i^{s+1} = L_i^s$.

   Let $x^{s+1} \in \{\langle i, z\rangle \mid z \in N\}$ be such that $x^{s+1} > n$.

   Go to stage $s+1$.

End stage s

It is easy to verify that L_i can be enumerated effectively in i. Fix i. We consider two cases in the definition of L_i.

Case 1: There exist infinitely many stages.

In this case $\mathbf{M}_i$ on canonical informant for L_i makes infinitely many mind changes.

Case 2: Stage s starts but does not end.

In this case $\mathbf{M}_i$ converges to the same grammar for both L_i and $L_i \cup \{x^s\}$ (which are distinct languages). Thus $\mathbf{M}_i$ is not reliable.

The above cases show that $\mathcal{L}$ is not **EInfRel**-identified by $\mathbf{M}_i$. ∎

Theorem 25 **TxtFin** $-$ **ETxtJRef** $\neq \emptyset$.

PROOF. The proof of this theorem is similar to that of Theorem 24. For each i, we will define below a nonempty language L_i with the following properties:

(a) $L_i \subseteq \{\langle i, n\rangle \mid n \in N\}$;

(b) a grammar for L_i can be obtained effectively in i.

We take $\mathcal{L} = \{L_i \mid i \in N\}$. Clearly, $\mathcal{L} \in \mathbf{TxtFin}$ (since a grammar for L_i can be found effectively from i). Further, we will have that $\mathbf{M}_i$ does not **ETxtJRef**-identify $\mathcal{L}$.

We will define L_i in stages below. Let L_i^s denote L_i defined before stage s. Let $L_i^0 = \{\langle i, 0\rangle\}$. Let σ_0 be such that content$(\sigma_0) = L_i^0$. Go to stage 0.

Stage s

1. Let T_1^s, T_2^s be texts extending σ_s, for $L_i^s \cup \{\langle i, 2s+1\rangle\}$ and $L_i^s \cup \{\langle i, 2s+2\rangle\}$ respectively.
2. Search for $n > |\sigma_s|$ such that either $\mathbf{M}_i(T_1^s[n]) \neq \mathbf{M}_i(\sigma_s)$, or $\mathbf{M}_i(T_2^s[n]) \neq \mathbf{M}_i(\sigma_s)$, or $\mathbf{M}_i(T_1^s[n]) =\perp$, or $\mathbf{M}_i(T_2^s[n]) =\perp$.

If and when such an n is found proceed to step 3.

3. Let n be as found in step 2. If $\mathbf{M}_i(T_1^s[n]) \neq \mathbf{M}_i(\sigma_s)$ or $\mathbf{M}_i(T_1^s[n]) =\perp$, then let $L_i^{s+1} = L_i^s \cup \{2s+1\}$, and $\sigma = T_1^s[n]$. Otherwise, let $L_i^{s+1} = L_i^s \cup \{2s+2\}$, and $\sigma = T_2^s[n]$.

   Let σ_{s+1} be an extension of σ such that content$(\sigma_{s+1}) = L_i^{s+1}$.

   Go to stage $s+1$.

End stage s

It is easy to verify that L_i can be enumerated effectively in i. Fix i. We consider two cases in the definition of L_i.

Case 1: There exist infinitely many stages.

In this case $T = \bigcup_s \sigma_s$ is a text for L_i. However $\mathbf{M}_i$ either makes infinitely many mind changes on T, or refutes T.

Case 2: Stage s starts but does not end.

In this case note that (i) content(T_1^s) and content(T_2^s) are both finite, (ii) for large enough n, $T_1^s[n]$ and $T_2^s[n]$ are not in $\mathbf{Cons}_{\mathcal{L}}$, (iii) $\mathbf{M}_i$ does not refute the texts T_1^s and T_2^s, and (iv) $\mathbf{M}_i$ fails to **TxtEx**-identify at least one of T_1^s and T_2^s (since $\mathbf{M}_i$ converges to the same grammar on both of them). Thus, $\mathbf{M}_i$ does not **ETxtJRef**-identify $\mathcal{L}$.

The above cases show that $\mathcal{L}$ is not **ETxtJRef**-identified by $\mathbf{M}_i$. Since i was arbitrary, it follows that $\mathcal{L} \notin$ **ETxtJRef**. ∎

The following theorem shows the advantages of identification with refutation over finite identification.

Theorem 26 $(\mathbf{TxtRef} \cap \mathbf{TxtJIRef} \cap \mathbf{InfJIRef}) - \mathbf{InfFin} \neq \emptyset$.

Proof. Let $\mathcal{L} = \{L \mid \mathrm{card}(L) \leq 2\}$. It is easy to verify that $\mathcal{L}$ witnesses the separation. ∎

The following theorem shows the advantages of justified refutation and reliable identification over the case when the learning machine has to refute all unidentified texts (informants).

Theorem 27 $(\mathbf{TxtRel} \cap \mathbf{TxtJIRef} \cap \mathbf{InfJIRef}) - \mathbf{InfRef} \neq \emptyset$.

Proof. It is easy to verify that **FIN** witnesses the separation. ∎

The following theorem shows the disadvantages of immediate refutation. Note that for learning indexed families of recursive languages, Lange and Watson have shown that **TxtJRef** = **TxtJIRef** and **InfJRef** = **InfJIRef**, and thus the following result does not hold for learning indexed families of recursive languages.

Theorem 28 *(a)* $\mathbf{TxtRef} - \mathbf{EInfJIRef} \neq \emptyset$.
(b) $\mathbf{TxtRef} - \mathbf{ETxtJIRef} \neq \emptyset$.

Proof. Let $\mathcal{L} = \{\{i\} \mid i \notin \mathbf{K}\}$. It is easy to verify that $\mathcal{L} \in$ **TxtRef**. We show that $\mathcal{L} \notin$ **ETxtJIRef**. A similar proof also shows that $\mathcal{L} \notin$ **EInfJIRef**. Suppose by way of contradiction that **M** **ETxtJIRef**-identifies $\mathcal{L}$. Then the following claim shows that $\overline{\mathbf{K}}$ is r.e., a contradiction. Thus $\mathcal{L} \notin$ **ETxtJIRef**.

Claim 29 $i \in \overline{\mathbf{K}} \Leftrightarrow (\exists \sigma \mid \mathrm{content}(\sigma) = \{i\})[\mathbf{M}(\sigma)\downarrow \neq \perp]$.

PROOF. Suppose $i \in \overline{\mathbf{K}}$. Then, since $\mathbf{M}$ **TxtEx**-identifies $\{i\} \in \mathcal{L}$, there must exist a σ such that content$(\sigma) = \{i\}$ and $\mathbf{M}(\sigma)\downarrow \neq \perp$. On the other hand suppose by way of contradiction that $i \notin \overline{\mathbf{K}}$, and σ is such that content$(\sigma) = \{i\}$ and $\mathbf{M}(\sigma)\downarrow \neq \perp$. Let T be an extension of σ such that $\mathbf{M}$ does not **TxtEx**-identify T (there exists such a T by Proposition 6). But then, since $\sigma \notin \mathbf{Cons}_{\mathcal{L}}$, by definition of **ETxtJIRef**, $\mathbf{M}(\sigma)$ must be equal to $\perp$; a contradiction. This proves the claim, and completes the proof of the theorem. ∎

In the context of learnability of indexed families, Lange and Watson (in their model of learning with refutation) had shown that immediate refutation is not a restriction, i.e. **TxtJRef** = **TxtJIRef** and **InfJRef** = **InfJIRef**. The following corollary shows that immediate refutation is a restriction in the context of learning general classes of r.e. languages! Note that this restriction holds for both extended and unextended models of justified refutation (for general classes of r.e. languages).

Corollary 30 *(a)* **InfJIRef** $\subset$ **InfJRef**.
(b) **TxtJIRef** $\subset$ **TxtJRef**.
(c) **EInfJIRef** $\subset$ **EInfJRef**.
(d) **ETxtJIRef** $\subset$ **ETxtJRef**.

The following theorem shows the advantages of justified refutation over reliable identification.

Theorem 31 (**TxtJIRef** $\cap$ **InfJIRef**) $-$ **EInfRel** $\neq \emptyset$.

PROOF. For $f \in \mathcal{R}$, let $L_f = \{\langle x, y\rangle \mid f(x) = y\}$. Let $\mathcal{L} = \{L_f \mid \varphi_{f(0)} = f\} \cup \{L \mid L \in \mathbf{FIN} \wedge (\exists x, y, z \mid y \neq z)[\langle x, y\rangle \in L \wedge \langle x, z\rangle \in L]\}$. It is easy to verify that $\mathcal{L} \in$ **TxtEx** (and thus **InfEx**). Since, $\mathbf{ICons}_{\mathcal{L}} =$ SEQ, it follows that $\mathcal{L} \in$ **TxtJIRef** $\cap$ **InfJIRef**. Essentially the proof in [CJNM94] to show that $\{f \mid \varphi_{f(0)} = f\}$ cannot be identified by a reliable machine (for function learning) translates to show that $\mathcal{L} \notin$ **EInfRel**. ∎

The following theorem shows the advantages of reliable identification over justified refutation.

Theorem 32 *(a)* **TxtRel** $-$ **ETxtJRef** $\neq \emptyset$.
(b) **TxtRel** $-$ **EInfJRef** $\neq \emptyset$.

PROOF. (a) Let $L_i = \{\langle i, x\rangle \mid x \in N\}$. Let $X_i = \{\sigma \in \text{SEG} \mid \emptyset \subset \text{content}(\sigma) \subseteq L_i \wedge \mathbf{M}_i(\sigma) = \perp\}$. Note that X_i is recursively enumerable (a grammar for which can be found effectively in i).

Let $\mathcal{L} = \{\text{content}(\sigma) \mid (\exists i)[\sigma \in X_i]\}$.

It is easy to verify that $\mathcal{L} \in$ **TxtRel**. Suppose by way of contradiction that $\mathbf{M}_i$ witnesses that $\mathcal{L} \in$ **ETxtJRef**. We consider two cases,

Case 1: X_i is not empty.

In this case let $\sigma \in X_i$. Let T be a text for content(σ) that extends σ. Now content$(T) \in \mathcal{L}$, but $\mathbf{M}_i$ refutes T.

Case 2: X_i is empty.

In this case, $\mathcal{L}$ does not contain any subset of L_i. Let L be a nonempty subset of L_i such that $\mathbf{M}_i$ does not **TxtEx**-identify L (by Proposition 6 there must exist such an L). Let T be a text for L. Now $\mathbf{M}_i$ does not **TxtEx**-identify T, nor does it refute T. However, for all initial segments σ of T such that content$(\sigma) \neq \emptyset$, $\sigma \notin \mathbf{Cons}_{\mathcal{L}}$.

From the above cases, it follows that $\mathbf{M}_i$ does not **ETxtJRef**-identify $\mathcal{L}$. Thus $\mathcal{L} \notin$ **ETxtJRef**.

(b) Let L_i be as defined in part (a).
Let $X_i = \{\sigma \in \mathrm{SEQ} \mid \emptyset \subset \mathrm{PosInfo}(\sigma) \subseteq L_i \ \wedge \ \mathbf{M}_i(\sigma) = \perp\}$.
Let $\mathcal{L} = \{\mathrm{PosInfo}(\sigma) \mid (\exists i)[\sigma \in X_i]\}$.
It can now be proved in a manner similar to part (a) that $\mathcal{L} \in \mathbf{TxtRel} - \mathbf{EInfJRef}$. ∎

The following theorem shows the advantages of having an informant over texts.

Theorem 33 $(\mathbf{InfRef} \cap \mathbf{InfJIRef}) - \mathbf{TxtEx} \neq \emptyset$*.*

Proof. $\mathbf{INIT} \cup \{N\}$ witnesses the separation. ∎

5 A Variation of Extended Justified Refutation Criteria

In the definitions for (extended) criteria of learning with justified refutation, we required the machines to either identify or (immediately) refute any text (informant) which did not contain a finite sample representative of the class, $\mathcal{L}$, being learned. For example, in **ETxtJRef**-identification we required that the machine either identify or refute every text which starts with an initial segment not in $\mathbf{Cons}_{\mathcal{L}}$. Alternatively, we could place such a restriction only for texts which are not representative of what the machine identifies (note that this gives more freedom to the machine). In other words, in the definitions for $\mathbf{ETxtJRef}, \mathbf{ETxtJIRef}, \mathbf{EInfJRef}, \mathbf{EInfJIRef}$, we could have taken $\{T[n] \mid n \in N$ and $\mathbf{M}$ $\mathbf{TxtEx}$-identifies $T\}$, instead of $\mathbf{Cons}_{\mathcal{L}}$ and $\{I[n] \mid n \in N$ and $\mathbf{M}$ $\mathbf{InfEx}$-identifies $I\}$, instead of $\mathbf{ICons}_{\mathcal{L}}$. Let these new classes formed be called $\mathbf{ETxtJRef}'$, $\mathbf{EInfJRef}'$, $\mathbf{ETxtJIRef}'$, $\mathbf{EInfJIRef}'$. Note that a similar change does not effect the classes **ETxtRef**, **ETxtRel**, **EInfRef**, **EInfRel**.

An easy to show interesting property of the classes $\mathbf{ETxtJRef}'$, $\mathbf{EInfJRef}'$, $\mathbf{ETxtJIRef}'$, $\mathbf{EInfJIRef}'$ is that they are closed under subset operation (i.e., if $\mathcal{L} \in \mathbf{ETxtJRef}'$, then every $\mathcal{L}' \subseteq \mathcal{L}$ is in $\mathbf{ETxtJRef}'$). Note that **ETxtJIRef**, **ETxtJRef**, **EInfJIRef**, **EInfJRef** are not closed under subset operation — this follows immediately from Theorem 32 and the fact that **FIN** belongs to each of these inference criteria.

We now show a result that $\mathbf{ETxtJIRef}'$ and $\mathbf{ETxtJRef}'$ obtain the full power of **TxtEx**! This is a surprising result given the results in [LW94] and this paper (**ETxtJRef** and **EInfJRef** do not even contain **TxtFin**, as shown in Theorem 24 and Theorem 25).

Theorem 34 *(a)* $\mathbf{TxtEx} = \mathbf{ETxtJRef}' = \mathbf{ETxtJIRef}'$*.*
(b) $\mathbf{EInfJRef}' = \mathbf{EInfJIRef}'$*.*

Proof. For part (a), it is enough to show that $\mathbf{TxtEx} \subseteq \mathbf{ETxtJIRef}'$. Consider any class $\mathcal{L} \in \mathbf{TxtEx}$. If $N \in \mathcal{L}$, then it immediately follows that $\mathcal{L} \in \mathbf{ETxtJIRef}'$ (since $\mathbf{Cons}_{\mathcal{L}} = \mathrm{SEG}$). If $N \notin \mathcal{L}$, then let $\mathcal{L}' = \mathcal{L} \cup \mathbf{INIT}$. It was shown by Fulk [Ful90] that, if $\mathcal{L} \in \mathbf{TxtEx}$ and $\mathcal{L}$ does not contain N, then $\mathcal{L}'$ as defined above is in **TxtEx**. Since $\mathcal{L}'$ contains a superset of every finite set, it follows that $\mathcal{L}' \in \mathbf{ETxtJIRef}'$ (since $\mathbf{Cons}_{\mathcal{L}'} = \mathrm{SEG}$). Part (a) now follows using the fact that $\mathbf{ETxtJIRef}'$ is closed under subset operation.

(b) It is sufficient to show that $\mathbf{EInfJRef}' \subseteq \mathbf{EInfJIRef}'$. Suppose $\mathbf{M}$ $\mathbf{EInfJRef}'$-identifies $\mathcal{L}$. We construct an $\mathbf{M}'$ which $\mathbf{EInfJIRef}'$-identifies $\mathcal{L}$. Let g denote a recursive function such that, for all finite sets S, $W_{g(S)} = S$. On any input $I[n]$, $\mathbf{M}'$ behaves as follows. If $\mathbf{M}(I[n]) \neq \perp$, then $\mathbf{M}'(I[n]) = \mathbf{M}(I[n])$ (this ensures that $\mathbf{M}'$ **InfEx**-identifies $\mathcal{L}$). If $\mathbf{M}(I[n]) = \perp$, then let m be the smallest number such that $\mathbf{M}(I[m]) = \perp$. If $\mathrm{PosInfo}(I[n]) = \mathrm{PosInfo}(I[m])$, then $\mathbf{M}'(I[n])$ outputs

$g(\text{PosInfo}(I[n]))$. Otherwise $\mathbf{M}'(I[n])$ outputs $\perp$. We claim that for every σ, either $\mathbf{M}'(\sigma) =\perp$ or there exists an extension I of σ such that $\mathbf{M}'$ **InfEx**-identifies I. So suppose $\mathbf{M}'(\sigma) \neq\perp$. We consider the following cases.

Case 1: $\mathbf{M}(\sigma) \neq\perp$.

If $\mathbf{M}$ **InfEx**-identifies some extension of σ, then clearly $\mathbf{M}'$ does too. So suppose that $\mathbf{M}$ does not **InfEx**-identify any extension of σ. This implies that $\mathbf{M}$ refutes every informant which begins with σ. Let I be an informant, extending σ, for PosInfo(σ). Let n be the least number such that $\mathbf{M}(I[n]) =\perp$. Note that $I[n]$ must be an extension of σ. It now follows from the definition of $\mathbf{M}'$ that $\mathbf{M}'$ **InfEx**-identifies I.

Case 2: $\mathbf{M}(\sigma) =\perp$.

Let τ be the smallest prefix of σ such that $\mathbf{M}'(\tau) =\perp$. It follows from the definition of $\mathbf{M}'$ that PosInfo(τ) = PosInfo(σ). Let I, extending σ, be an informant for PosInfo(σ). It follows from the definition of $\mathbf{M}'$ that $\mathbf{M}'(I)$ is a grammar for PosInfo(τ) = PosInfo(I).

From the above cases, it follows that $\mathbf{M}'$ **EInfJIRef**$'$-identifies $\mathcal{L}$. ∎

However, unlike the case for texts, **EInfJRef**$'$, is not equal to **InfEx**.

Theorem 35 **TxtEx** $-$ **EInfJRef**$'$ $\neq \emptyset$.

Proof. First we prove the following lemma. Note that if $\mathbf{M}$ **EInfJRef**$'$-identifies $\mathcal{L}$, then for each finite information sequence σ, it must satisfy either (a) or (b) in the statement of lemma.

Lemma 36 *Suppose* $\mathbf{M}$ *is such that for all finite information sequences* σ*, at least one of the following two properties is satisfied:*

(a) $\mathbf{M}$ **InfEx**-*identifies some informant extending* σ*, or*

(b) $\mathbf{M}$ *refutes all infinite information sequences* I *extending* σ.

Then, for every finite information sequence τ*, one can find (effectively in* τ *and* $\mathbf{M}$*) a grammar* $g(\tau, \mathbf{M})$ *such that* $W_{g(\tau,\mathbf{M})} \supseteq$ *PosInfo*(τ)*, and either*

(c) There exists a γ *extending* τ *such that* $\mathbf{M}(\gamma) =\perp$*, or*

(d) $\mathbf{M}$ *diverges on some informant for* $W_{g(\tau,\mathbf{M})}$.

Proof. Suppose $\mathbf{M}$ is given such that for each finite information sequence σ, either (a) or (b) holds. Let τ be such that (c) fails. Then, we claim that, for each γ extending τ, there exists a γ''' extending γ such that $\mathbf{M}(\gamma) \neq \mathbf{M}(\gamma''')$. To see this, consider any γ extending τ. Let γ' and γ'' be two extensions of γ such that $\text{PosInfo}(\gamma') \cap \text{NegInfo}(\gamma'') \neq \emptyset$ (i.e. γ' and γ'' are inconsistent with each other). Now, since (b) is not true for $\sigma = \gamma'$ or $\sigma = \gamma''$, we have from (a) that $\mathbf{M}$ **InfEx**-identifies some informant extending γ' as well as some informant extending γ''. Thus, there exists a γ''' extending γ such that $\mathbf{M}(\gamma) \neq \mathbf{M}(\gamma''')$. Hence, for every γ extending τ, there exists a γ''' extending γ such that $\mathbf{M}(\gamma) \neq \mathbf{M}(\gamma''')$. This is what our construction of g will utilize. We give below a description of $W_{g(\tau,\mathbf{M})}$ (note that g can be defined using s-m-n theorem).

$W_{g(\tau,\mathbf{M})}$

1. Let $\tau_0 = \tau$.
   Enumerate PosInfo(τ) into $W_{g(\tau,\mathbf{M})}$.
   Go to stage 0.
2. Stage s

Search for γ extending τ_s such that $\mathbf{M}(\gamma) \neq \mathbf{M}(\tau_s)$.
Let τ_{s+1} be an extension of γ such that $\mathrm{PosInfo}(\tau_{s+1}) \cup \mathrm{NegInfo}(\tau_{s+1}) \supseteq \{x \mid x \leq s\}$.
Enumerate $\mathrm{PosInfo}(\tau_{s+1})$ in $W_{g(\tau,\mathbf{M})}$.
Go to stage $s+1$.

End stage s

End

Note that if $\mathbf{M}$ is as given in the lemma and τ is such that (c) fails, then, based on analysis above, there will be infinitely many stages in the construction of $W_{g(\tau,\mathbf{M})}$. Thus, $\bigcup_i \tau_i$ is an informant for $W_{g(\tau,\mathbf{M})}$ on which $\mathbf{M}$ makes infinitely many mind changes. This proves the lemma. □

We now proceed with the proof of the theorem.

Let τ_i denote a finite information sequence (obtained effectively from i) such that $\mathrm{PosInfo}(\tau_i) = \{i+1\}$ and $\mathrm{NegInfo}(\tau_i) = \{x \mid x \leq i\}$.

Let $S_i = \{\gamma \mid \tau_i \preceq \gamma \ \wedge \ \mathbf{M}_i(\gamma) = \perp\}$.

Let $\mathcal{L} = \{W_{g(\tau_i,\mathbf{M}_i)} \mid S_i = \emptyset\} \cup \{\mathrm{PosInfo}(\gamma) \mid \gamma \in S_i\}$.

It is easy to verify that $\mathcal{L} \in \mathbf{TxtEx}$. However, if S_i is not empty, then $\mathbf{M}_i$ refutes $\gamma \in S_i$, even though $\mathrm{PosInfo}(\gamma) \in \mathcal{L}$. If S_i is empty, then either $\mathbf{M}_i$ does not behave properly (i.e. $\mathbf{M}_i$, for some σ, fails to satisfy both conditions (a) and (b) in the statement of Lemma 36) or does not **InfEx**-identify $W_{g(\tau_i,\mathbf{M}_i)} \in \mathcal{L}$.

It follows that $\mathcal{L} \notin \mathbf{EInfJRef}'$. ∎

Corollary 37 $\mathbf{TxtJIRef} - \mathbf{EInfJRef}' \neq \emptyset$.

Proof. Note that no language in the class $\mathcal{L}$ defined in the proof of Theorem 35 contains 0. Thus, $\mathcal{L} \cup \{N\} \in \mathbf{TxtEx}$. However, $\mathcal{L} \cup \{N\} \notin \mathbf{EInfJRef}'$ using proof identical to the proof of Theorem 35. ∎

The following theorem, however, shows that $\mathbf{EInfJRef}'$ contains **InfRel**.

Theorem 38 $\mathbf{InfRel} \subseteq \mathbf{EInfJIRef}'$

Proof. Note that **InfRel** is closed under finite union. Also $\mathbf{FIN} \in \mathbf{InfRel}$. Now suppose $\mathcal{L} \in \mathbf{InfRel}$. Thus $(\mathcal{L} \cup \mathbf{FIN}) \in \mathbf{InfRel} \subseteq \mathbf{InfEx}$. It follows that $(\mathcal{L} \cup \mathbf{FIN}) \in \mathbf{EInfJIRef}'$ (since $\mathbf{ICons}_{\mathcal{L}\cup\mathbf{FIN}} = \mathrm{SEQ}$). Now, $\mathbf{EInfJIRef}'$ is closed under subset operation, and thus it follows that $\mathcal{L} \in \mathbf{EInfJIRef}'$. ∎

6 Conclusions

Mukouchi and Arikawa modeled a learning situation in which the learner is expected to refute texts which are not representative of $\mathcal{L}$, the class of languages being identified. Lange and Watson extended this model to consider justified refutation in which the learner is expected to refute texts only if it contains a finite sample unrepresentative of the class $\mathcal{L}$. Both the above studies were in the context of indexed families of recursive languages. In this paper we extended this study in two directions. Firstly, we considered general classes of recursively enumerable languages. Secondly, we allowed the machine to either identify or refute the unrepresentative texts (respectively, texts containing finite unrepresentative samples). We observed some surprising differences between our results and the results obtained for learning indexed families by Lange and Watson. For example, in the context of learning indexed families of recursive languages, Lange and Watson (in their model) showed that **TxtJIRef** = **TxtJRef** (i.e. requiring machines to refute as soon as the initial segment becomes unrepresentative of $\mathcal{L}$, is not a restriction). Similar result was also shown by them for learning from informants. We showed that requiring immediate refutation is a restriction if we consider general classes of r.e. languages (in both our (extended) and Lange and Watson's models of justified refutation, and for learning from texts as well as informants). We also considered a variation of our model in which "unrepresentative" is with respect to what a machine identifies and not with respect to the class $\mathcal{L}$. In this variation, for learning from texts, (immediate) justified refutation model has the same power as **TxtEx** — a surprising result in the context of results in [LW94] and other results in this paper. However, in the context of learning from informants, even this variation fails to capture the power of **InfEx**.

It would be useful to find interesting characterizations of the different inference classes studied in this paper. An anonymous referee suggested the following problems. When we do not require immediate refutation (as in **TxtJRef**) the delay in refuting the text may be arbitrarily large. It would be interesting to study any hierarchy that can be formed by "quantifying" the delay. Note that if one just considers the number of excess data points needed before refuting, then the hierarchy collapses — except for the $*$ (unbounded but finite) case. As extensions of criteria considered in this paper, one could consider the situation when a machine approximately identifies [KY95, KY97, Muk94] a text T in the cases when it doesn't identify or refute a text.

7 Acknowledgments

We thank Arun Sharma for helpful discussions and comments. Anonymous referees provided several comments which improved the presentation of this paper.

References

[Blu67] M. Blum. A machine-independent theory of the complexity of recursive functions. *Journal of the ACM*, 14:322–336, 1967.

[CJNM94] J. Case, S. Jain, and S. Ngo Manguelle. Refinements of inductive inference by Popperian and reliable machines. *Kybernetika*, 30:23–52, 1994.

[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.

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

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

[KY95] S. Kobayashi and T. Yokomori. On approximately identifying concept classes in the limit. In *Algorithmic Learning Theory: Sixth International Workshop (ALT '95)*, volume 997 of *Lecture Notes in Artificial Intelligence*, pages 298–312. Springer-Verlag, 1995.

[KY97] S. Kobayashi and T. Yokomori. Learning approximately regular languages with reversible languages. *Theoretical Computer Science A*, 174:251–257, 1997.

[LW94] S. Lange and P. Watson. Machine discovery in the presence of incomplete or ambiguous data. In S. Arikawa and K. Jantke, editors, *Algorithmic learning theory: Fourth International Workshop on Analogical and Inductive Inference (AII '94) and Fifth International Workshop on Algorithmic Learning Theory (ALT '94)*, volume 872 of *Lecture Notes in Artificial Intelligence*, pages 438–452. Springer-Verlag, 1994.

[MA93] Y. Mukouchi and S. Arikawa. Inductive inference machines that can refute hypothesis spaces. In K.P. Jantke, S. Kobayashi, E. Tomita, and T. Yokomori, editors, *Algorithmic Learning Theory: Fourth International Workshop (ALT '93)*, volume 744 of *Lecture Notes in Artificial Intelligence*, pages 123–136. Springer-Verlag, 1993.

[MA95] Y. Mukouchi and S. Arikawa. Towards a mathematical theory of machine discovery from facts. *Theoretical Computer Science A*, 137:53–84, 1995.

[Min76] E. Minicozzi. Some natural properties of strong identification in inductive inference. *Theoretical Computer Science*, pages 345–360, 1976.

[Muk94] Y. Mukouchi. Inductive inference of an approximate concept from positive data. In S. Arikawa and K. Jantke, editors, *Algorithmic learning theory: Fourth International Workshop on Analogical and Inductive Inference (AII '94) and Fifth International Workshop on Algorithmic Learning Theory (ALT '94)*, volume 872 of *Lecture Notes in Artificial Intelligence*, pages 484–499. Springer-Verlag, 1994.

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

[Sha98] A. Sharma. A note on batch and incremental learnability. *Journal of Computer and System Sciences*, 1998. to appear.