Computability Theory
Learning Programs to Fit/Predict Data & Machine Self-Reference

John Case

Computer and Information Sciences Department
University of Delaware
Newark, DE 19716 USA

Email: case@cis.udel.edu

SIGNewGrad 2012, 114 Gore
Sample of Computational Learning Theory Results.

- Today’s Sample: Applicable to Cognitive Science
  [CCJS07, CCJS08, BCM+08, CM08, CK10b, CK10a].
- I’m also interested in other results applicable to Philosophy of Science
  [CS83, CJNM94, Cas07, Cas12] and empirical Machine Learning
  [CJO+00, CJK+01, COSS02, CJM+06, CJ10].

- My Theory project re Machine Self-Reference
  [CM09a, CM12, CM09b, CM11].
Sample of Computational Learning Theory Results.

Today’s Sample: Applicable to Cognitive Science [CCJS07, CCJS08, BCM+08, CM08, CK10b, CK10a].

I’m also interested in other results applicable to Philosophy of Science [CS83, CJNM94, Cas07, Cas12] and empirical Machine Learning [CJO+00, CJK+01, COSS02, CJM+06, CJ10].

My Theory project re Machine Self-Reference [CM09a, CM12, CM09b, CM11].
Sample of Computational Learning Theory Results.

Today’s Sample: Applicable to Cognitive Science [CCJS07, CCJS08, BCM+08, CM08, CK10b, CK10a].

I’m also interested in other results applicable to Philosophy of Science [CS83, CJNM94, Cas07, Cas12] and empirical Machine Learning [CJO+00, CJK+01, COSS02, CJM+06, CJ10].

My Theory project re Machine Self-Reference [CM09a, CM12, CM09b, CM11].
Sample of Computational Learning Theory Results.

Today’s Sample: Applicable to Cognitive Science [CCJS07, CCJS08, BCM^{+}08, CM08, CK10b, CK10a].
I’m also interested in other results applicable to Philosophy of Science [CS83, CJNM94, Cas07, Cas12] and empirical Machine Learning [CJO^{+}00, CJK^{+}01, COSS02, CJM^{+}06, CJ10].

My Theory project re Machine Self-Reference [CM09a, CM12, CM09b, CM11].
Sample of Computational Learning Theory Results.

Today’s Sample: Applicable to Cognitive Science [CCJS07, CCJS08, BCM+08, CM08, CK10b, CK10a].
I’m also interested in other results applicable to Philosophy of Science [CS83, CJNM94, Cas07, Cas12] and empirical Machine Learning [CJO+00, CJK+01, COSS02, CJM+06, CJ10].

My Theory project re Machine Self-Reference [CM09a, CM12, CM09b, CM11].
U-Shaped Learning

- Learn, Unlearn, Relearn. Occurs in child development re, e.g., verb regularization & understanding of various (Piaget-like) conservation principles, e.g., temperature & weight conservation & interaction bet. object tracking/object permanence.

- Irregular Verb Example: Child first uses *spoke*, correct past tense of irregular verb *to speak*. Then child ostensibly overregularizes incorrectly using *spaked*. Lastly, child returns to using *spoke*.

- Concern of Prior Literature: How model U-shaped learning? E.g., lang. learn., by gen. rules vs. tables of exceptions?

- My Interest: Is U-shaped learning an unnecessary accident of human evolution or is U-shaped learning advantageous in that some classes of tasks can be learned in U-shaped way, but not otherwise?
U-Shaped Learning

- Learn, Unlearn, Relearn. Occurs in child development re, e.g., verb regularization & understanding of various (Piaget-like) conservation principles, e.g., temperature & weight conservation & interaction bet. object tracking/object permanence.

- Irregular Verb Example: Child first uses *spoke*, correct past tense of irregular verb *to speak*. Then child ostensibly overregularizes incorrectly using *spaked*. Lastly, child returns to using *spoke*.

- Concern of Prior Literature: How model U-shaped learning? E.g., lang. learn., by gen. rules vs. tables of exceptions?

- My Interest: Is U-shaped learning an unnecessary accident of human evolution or is U-shaped learning advantageous in that some classes of tasks can be learned in U-shaped way, but not otherwise?
U-Shaped Learning

- Learn, Unlearn, Relearn. Occurs in child development re, e.g., verb regularization & understanding of various (Piaget-like) conservation principles, e.g., temperature & weight conservation & interaction bet. object tracking/object permanence.

- Irregular Verb Example: Child first uses *spoke*, correct past tense of irregular verb *to speak*. Then child ostensibly overregularizes incorrectly using *spaked*. Lastly, child returns to using *spoke*.

- Concern of Prior Literature: How model U-shaped learning? E.g., lang. learn., by gen. rules vs. tables of exceptions?

- My Interest: Is U-shaped learning an unnecessary accident of human evolution or is U-shaped learning advantageous in that some classes of tasks can be learned in U-shaped way, but not otherwise?
U-Shaped Learning

- Learn, Unlearn, Relearn. Occurs in child development re, e.g., verb regularization & understanding of various (Piaget-like) conservation principles, e.g., temperature & weight conservation & interaction between object tracking/object permanence.

- Irregular Verb Example: Child first uses *spoke*, correct past tense of irregular verb *to speak*. Then child ostensibly overregularizes incorrectly using *spaked*. Lastly, child returns to using *spoke*.

- Concern of Prior Literature: How model U-shaped learning? E.g., lang. learn., by gen. rules vs. tables of exceptions?

- My Interest: Is U-shaped learning an unnecessary accident of human evolution or is U-shaped learning advantageous in that some classes of tasks can be learned in U-shaped way, but not otherwise?
U-Shaped Learning

- Learn, Unlearn, Relearn. Occurs in child development re, e.g., verb regularization & understanding of various (Piaget-like) conservation principles, e.g., temperature & weight conservation & interaction bet. object tracking/object permanence.

- Irregular Verb Example: Child first uses *spoke*, correct past tense of irregular verb *to speak*. Then child ostensibly overregularizes incorrectly using *spaked*. Lastly, child returns to using *spoke*.

- Concern of Prior Literature: How model U-shaped learning? E.g., lang. learn., by gen. rules vs. tables of exceptions?

- My Interest: Is U-shaped learning an unnecessary accident of human evolution or is U-shaped learning advantageous in that some classes of tasks can be learned in U-shaped way, but not otherwise?
Formal Definitions

- $T(0), \ T(1), \ldots \xrightarrow{\text{In}} M \xrightarrow{\text{Out}} g_0, g_1, \ldots, \!
\!
| \ g_t, \ldots$
- Criteria for: some $M$ successfully learns every language $L$ in class $\mathcal{L}$. Suppose: $N^+ = \{1, 2, \ldots\}$; $b \in (N^+ \cup \{\ast\})$; $x \leq \ast$ means $x < \infty$; $T$ is a text for $L \overset{\text{def}}{=} \{T(0), T(1), \ldots\} = L$; & $W_g \overset{\text{def}}{=} \text{lang. generated by grammar } g$ — $W_g$ is behavior of $g$.
- $\mathcal{L} \in \text{TxtFex}_b$: ($\exists M)(\forall L \in \mathcal{L})(\forall T \text{ for } L)(\exists t) \ [g_t, g_{t+1}, \ldots$
\!
\!
\!each generates $L$ & $\text{card}(\{g_t, g_{t+1}, \ldots\}) \leq b]$.
- $\mathcal{L} \in \text{TxtBc}$:
\!
\!
\!($\exists M)(\forall L \in \mathcal{L})(\forall T \text{ for } L)(\exists t) \ [g_t, g_{t+1}, \ldots$
\!
\!each generates $L]$.
- Suppose $C \in \{\text{TxtFex}_b, \text{TxtBc}\}$. Then, $\mathcal{L} \in \text{NonUC}$: ($\exists M$ witnessing $\mathcal{L} \in C)(\forall L \in \mathcal{L})(\forall T \text{ for } L)(\forall i, j, k | i < j < k)[W_{g_i} = W_{g_k} = L \Rightarrow W_{g_j} = L]$. Non U-shaped learners never abandon correct behaviors $\in \mathcal{L}$ and return to them.
Formal Definitions

- \( T(0), T(1), \ldots \xrightarrow{\text{In}} M \xrightarrow{\text{Out}} g_0, g_1, \ldots, | g_t, \ldots \)
- Criteria for: some \( M \) successfully learns every language \( L \) in class \( \mathcal{L} \).
  Suppose: \( N^+ = \{1, 2, \ldots\}; \ b \in (N^+ \cup \{\ast\}); \ x \leq \ast \) means \( x < \infty \); \( T \)
  is a \text{text} for \( L \) \( \overset{\text{def}}{=} \{ T(0), T(1), \ldots \} = L; \) & \( W_g \overset{\text{def}}{=} \) lang. generated by
  grammar \( g \) — \( W_g \) is behavior of \( g \).
- \( \mathcal{L} \in \text{TxtFx}_b \):
  \( (\exists M)(\forall L \in \mathcal{L})(\forall T \text{ for } L)(\exists t) \ [g_t, g_{t+1}, \ldots \)
  each generates \( L \) & \( \text{card}(\{g_t, g_{t+1}, \ldots\}) \leq b \]. \( \text{TxtEx} \overset{\text{def}}{=} \text{TxtFx}_1 \).
- \( \mathcal{L} \in \text{TxtBc} \):
  \( (\exists M)(\forall L \in \mathcal{L})(\forall T \text{ for } L)(\exists t) \ [g_t, g_{t+1}, \ldots \)
  each generates \( L \].
- Suppose \( C \in \{ \text{TxtFx}_b, \text{TxtBc} \} \). Then, \( \mathcal{L} \in \text{NonUC} \):
  \( (\exists M \text{ witnessing } \mathcal{L} \in C)(\forall L \in \mathcal{L})(\forall T \text{ for } L)(\forall i, j, k \mid \)
  \( i < j < k \)[\( W_{g_i} = W_{g_k} = L \Rightarrow W_{g_j} = L \)]. Non-U-shaped
  learners never abandon correct behaviors \( \in \mathcal{L} \) and return to them.
Formal Definitions

- \( T(0), T(1), \ldots \xrightarrow{\text{In}} M \xrightarrow{\text{Out}} g_0, g_1, \ldots, | g_t, \ldots \)

- Criteria for: some \( M \) successfully learns every language \( L \) in class \( \mathcal{L} \).
  Suppose: \( N^+ = \{1, 2, \ldots\} \); \( b \in (N^+ \cup \{\ast\}) \); \( x \leq \ast \) means \( x < \infty \); \( T \) is a text for \( L \xleftrightarrow{\text{def}} \{T(0), T(1), \ldots\} = L \); \( \mathcal{W}_g \equiv \text{lang. generated by grammar } g \). — \( \mathcal{W}_g \) is behavior of \( g \).

- \( \mathcal{L} \in \text{TxtFex}_b \):
  \( (\exists M)(\forall L \in \mathcal{L})(\forall T \text{ for } L)(\exists t) [g_t, g_{t+1}, \ldots \text{ each generates } L \& \text{ card}({g_t, g_{t+1}, \ldots}) \leq b] \). \( \text{TxtEx} \overset{\text{def}}{=} \text{TxtFex}_1 \).

- \( \mathcal{L} \in \text{TxtBc} \):
  \( (\exists M)(\forall L \in \mathcal{L})(\forall T \text{ for } L)(\exists t) [g_t, g_{t+1}, \ldots \text{ each generates } L] \).

- Suppose \( C \in \{\text{TxtFex}_b, \text{TxtBc}\} \). Then,
  \( \mathcal{L} \in \text{NonUC} \): (\exists M \text{ witnessing } \mathcal{L} \in C)(\forall L \in \mathcal{L})(\forall T \text{ for } L)(\forall i, j, k | i < j < k)\mathcal{W}_g_i = \mathcal{W}_g_k = L \Rightarrow \mathcal{W}_g_j = L \). Non U-shaped learners never abandon correct behaviors \( \in \mathcal{L} \) and return to them.
Formal Definitions

- \( T(0), T(1), \ldots \xrightarrow{\text{In}} M \xrightarrow{\text{Out}} g_0, g_1, \ldots, | g_t, \ldots \)

- Criteria for: some \( M \) successfully learns every language \( L \) in class \( \mathcal{L} \). Suppose: \( N^+ = \{1, 2, \ldots\} \); \( b \in (N^+ \cup \{\ast\}) \); \( x \leq \ast \) means \( x < \infty \); \( T \) is a text for \( L \) \( \overset{\text{def}}{=} \{ T(0), T(1), \ldots \} = L \); & \( W_g \overset{\text{def}}{=} \text{lang. generated by grammar } g \rightleftharpoons W_g \) is behavior of \( g \).

- \( \mathcal{L} \in \text{TxtFex}_b \):
  \[
  (\exists M)(\forall L \in \mathcal{L})(\forall T \text{ for } L)(\exists t) \ [g_t, g_{t+1}, \ldots \text{ each generates } L \text{ & } \text{card}(\{g_t, g_{t+1}, \ldots\}) \leq b]. \]

- \( \mathcal{L} \in \text{TxtBc} \):
  \[
  (\exists M)(\forall L \in \mathcal{L})(\forall T \text{ for } L)(\exists t) \ [g_t, g_{t+1}, \ldots \text{ each generates } L]. 
  \]

- Suppose \( C \in \{\text{TxtFex}_b, \text{TxtBc}\} \). Then, \( \mathcal{L} \in \text{NonUC} \): (\exists M \text{ witnessing } \mathcal{L} \in C)(\forall L \in \mathcal{L})(\forall T \text{ for } L)(\forall i, j, k \mid i < j < k)[W_{g_i} = W_{g_k} = L \Rightarrow W_{g_j} = L]. \) Non U-shaped learners never abandon correct behaviors \( \in \mathcal{L} \) and return to them.
Formal Definitions

- \( T(0), T(1), \ldots \xrightarrow{\text{In}} M \xrightarrow{\text{Out}} g_0, g_1, \ldots, | g_t, \ldots \)
- Criteria for: some \( M \) successfully learns every language \( L \) in class \( \mathcal{L} \).
  Suppose: \( N^+ = \{1, 2, \ldots\} \); \( b \in (N^+ \cup \{\star\}) \); \( x \leq \star \) means \( x < \infty \); \( T \) is a text for \( L \) def \( \{ T(0), T(1), \ldots \} = L \); \& \( W_g \) def = lang. generated by grammar \( g \) — \( W_g \) is behavior of \( g \).
- \( \mathcal{L} \in \text{TxtFex}_b \):
  \( (\exists M)(\forall L \in \mathcal{L})(\forall T \text{ for } L)(\exists t) [g_t, g_{t+1}, \ldots \text{ each generates } L \& \text{card(}\{g_t, g_{t+1}, \ldots\}\text{)} \leq b] \). \( \text{TxtEx} \) def = \( \text{TxtFex}_1 \).
- \( \mathcal{L} \in \text{TxtBc} \):
  \( (\exists M)(\forall L \in \mathcal{L})(\forall T \text{ for } L)(\exists t) [g_t, g_{t+1}, \ldots \text{ each generates } L] \).
- Suppose \( C \in \{\text{TxtFex}_b, \text{TxtBc}\} \). Then, \( \mathcal{L} \in \text{NonUC} \): \( (\exists M \text{ witnessing } \mathcal{L} \in C)(\forall L \in \mathcal{L})(\forall T \text{ for } L)(\forall i, j, k | i < j < k)[W_{g_i} = W_{g_k} = L \Rightarrow W_{g_j} = L] \). Non U-shaped learners never abandon correct behaviors \( \in \mathcal{L} \) and return to them.
Formal Definitions

- \( T(0), T(1), \ldots \xrightarrow{\text{In}} M \xrightarrow{\text{Out}} g_0, g_1, \ldots, | g_t, \ldots \)

- Criteria for: some \( M \) successfully learns every language \( L \) in class \( \mathcal{L} \).
  Suppose: \( N^+ = \{1, 2, \ldots\}; b \in (N^+ \cup \{\ast\}); x \leq \ast \) means \( x < \infty \); \( T \) is a text for \( L \equiv \{ T(0), T(1), \ldots \} = L \); & \( W_g \equiv \) lang. generated by grammar \( g \) — \( W_g \) is behavior of \( g \).

- \( \mathcal{L} \in \text{TxtFex}_b \):
  \( (\exists M)(\forall L \in \mathcal{L})(\forall T \text{ for } L)(\exists t) [g_t, g_{t+1}, \ldots \text{ each generates } L \text{ & } \text{card}([g_t, g_{t+1}, \ldots]) \leq b] \). \( \text{TxtEx} \equiv \text{TxtFex}_1 \).

- \( \mathcal{L} \in \text{TxtBc} \):
  \( (\exists M)(\forall L \in \mathcal{L})(\forall T \text{ for } L)(\exists t) [g_t, g_{t+1}, \ldots \text{ each generates } L] \).

- Suppose \( C \in \{ \text{TxtFex}_b, \text{TxtBc} \} \). Then,
  \( \mathcal{L} \in \text{NonUC} \): \( (\exists M \text{ witnessing } \mathcal{L} \in C)(\forall L \in \mathcal{L})(\forall T \text{ for } L)(\forall i, j, k | i < j < k)[W_{g_i} = W_{g_k} = L \Rightarrow W_{g_j} = L] \). Non U-shaped learners never abandon correct behaviors \( \in \mathcal{L} \) and return to them.
The transitive closure of the following inclusions (\(\rightarrow\)) hold AND no other inclusions hold.

\[
\begin{align*}
\text{NonUTxtEx} & \rightarrow \text{TxtEx} \\
= & \text{TxtEx} \\
= & \text{NonUTxtFex}_b \\
\text{NonUTxtBc} & \rightarrow \text{TxtBc} \\
\end{align*}
\]

E.g., from the above, there is some \(L \in (\text{TxtFex}_3 - \text{NonUTxtBc})\)! This same \(L\) then cannot be \(\in \text{NonUTxtFex}_*\) — else, it would, then, be in \(\text{NonUTxtBc}\). This \(L\) does employ interplay between finite sets of exceptions & general rules.
The transitive closure of the following inclusions (\(\rightarrow\)) hold AND no other inclusions hold.

\[
\begin{align*}
\text{NonUTxtBc} & \rightarrow \text{TtxtBc} \\
\text{NonUTxtEx} & = \text{TtxtEx} \\
& \rightarrow \text{TtxtFex}_2 \rightarrow \text{TtxtFex}_3 \ldots \rightarrow \text{TtxtFex}_* \\
= \text{NonUTxtFex}_b
\end{align*}
\]

E.g., from the above, there is some \(\mathcal{L} \in (\text{TtxtFex}_3 - \text{NonUTxtBc})\)! This same \(\mathcal{L}\) then cannot be \(\in \text{NonUTxtFex}_*\) — else, it would, then, be in \(\text{NonUTxtBc}\). This \(\mathcal{L}\) does employ interplay between finite sets of exceptions & general rules.
Main Results and A Question

● Main Results:
  ● From NonUTxtBc → TxtBc, U-shaped learning is needed for some class in TxtBc.
  ● From NonUTxtEx = TxtEx, U-shaped learning is not needed for TxtEx learning, i.e., for learning ONE successful grammar in limit.
  ● From NonUTxtFex∗ → TxtFex₂, U-shaped learning is needed for some class in TxtFex₂, even if allow ∗ grammars in limit but, from TxtFex₂ → NonUTxtBc, is not needed if allow infinitely many grammars in limit.
  ● From the reasoning after the prior frame's diagram, exists $L \in (TxtFex₂ \setminus (NonUTxtFex∗ \cup NonUTxtBc))$. In particular, U-shaped learning IS needed for this $L \in TxtFex₂$ — even if allow infinitely many grammars in limit!

● Question: Does the class of tasks humans must learn to be competitive in the genetic marketplace, like this latter $L$, necessitate U-shaped learning?
Main Results and A Question

Main Results:

- From \( \text{NonUTxtBc} \rightarrow \text{TxtBc} \), U-shaped learning is needed for some class in \( \text{TxtBc} \).
- From \( \text{NonUTxtEx} = \text{TxtEx} \), U-shaped learning is not needed for \( \text{TxtEx} \) learning, i.e., for learning ONE successful grammar in limit.
- From \( \text{NonUTxtFex}^* \rightarrow \text{TxtFex}_2 \), U-shaped learning is needed for some class in \( \text{TxtFex}_2 \) even if allow \( ^* \) grammars in limit but, from \( \text{TxtFex}_2 \rightarrow \text{NonUTxtBc} \), is not needed if allow infinitely many grammars in limit.
- From the reasoning after the prior frame's diagram, exists \( \mathcal{L} \in (\text{TxtFex}_3 - (\text{NonUTxtFex}^* \cup \text{NonUTxtBc})) \); in particular, U-shaped learning IS needed for this \( \mathcal{L} \in \text{TxtFex}_3 \) — even if allow infinitely many grammars in limit!

Question: Does the class of tasks humans must learn to be competitive in the genetic marketplace, like this latter \( \mathcal{L} \), necessitate U-shaped learning?
Main Results and A Question

- **Main Results:**
  - From $\text{NonUTxtBc} \rightarrow \text{TxtBc}$, U-shaped learning is needed for some class in $\text{TxtBc}$.
  - From $\text{NonUTxtEx} = \text{TxtEx}$, U-shaped learning is not needed for $\text{TxtEx}$ learning, i.e., for learning ONE successful grammar in limit.
  - From $\text{NonUTxtFex}_* \rightarrow \text{TxtFex}_2$, U-shaped learning is needed for some class in $\text{TxtFex}_2$ even if allow $*$ grammars in limit but, from $\text{TxtFex}_2 \rightarrow \text{NonUTxtBc}$, is not needed if allow infinitely many grammars in limit.
  - From the reasoning after the prior frame’s diagram, exists $\mathcal{L} \in (\text{TxtFex}_3 - (\text{NonUTxtFex}_* \cup \text{NonUTxtBc}))$; in particular, U-shaped learning IS needed for this $\mathcal{L} \in \text{TxtFex}_3$ — even if allow infinitely many grammars in limit.

- **Question:** Does the class of tasks humans must learn to be competitive in the genetic marketplace, like this latter $\mathcal{L}$, necessitate U-shaped learning?
Main Results and A Question

- **Main Results:**
  - From **NonUTxtBc → TxtBc**, U-shaped learning is needed for some class in **TxtBc**.
  - From **NonUTxtEx = TxtEx**, U-shaped learning is not needed for **TxtEx** learning, i.e., for learning ONE successful grammar in limit.
  - From **NonUTxtFex∗ → TxtFex2**, U-shaped learning is needed for some class in **TxtFex2** even if allow ∗ grammars in limit but, from **TxtFex2 → NonUTxtBc**, is not needed if allow infinitely many grammars in limit.
  - From the reasoning after the prior frame’s diagram, exists \( \mathcal{L} \in (TxtFex_3 - (NonUTxtFex∗ \cup NonUTxtBc)) \); in particular, U-shaped learning IS needed for this \( \mathcal{L} \in TxtFex_3 \) — even if allow infinitely many grammars in limit!

- **Question:** Does the class of tasks humans must learn to be competitive in the genetic marketplace, like this latter \( \mathcal{L} \), necessitate U-shaped learning?
Main Results and A Question

- **Main Results:**
  - From **NonUTxtBc** → **TxtBc**, U-shaped learning is needed for some class in **TxtBc**.
  - From **NonUTxtEx** = **TxtEx**, U-shaped learning is not needed for **TxtEx** learning, i.e., for learning ONE successful grammar in limit.
  - From **NonUTxtFex_** → **TxtFex_2**, U-shaped learning is needed for some class in **TxtFex_2** even if allow * grammars in limit but, from **TxtFex_2** → **NonUTxtBc**, is not needed if allow infinitely many grammars in limit.
  - From the reasoning after the prior frame’s diagram, exists **L ∈ (TxtFex_3 – (NonUTxtFex_ * U NonUTxtBc))**; in particular, U-shaped learning IS needed for this **L ∈ TxtFex_3** — even if allow infinitely many grammars in limit!

- **Question:** Does the class of tasks humans must learn to be competitive in the genetic marketplace, like this latter **L**, necessitate U-shaped learning?
Main Results and A Question

Main Results:

- From $\text{NonUTxtBc} \rightarrow \text{TxtBc}$, U-shaped learning is needed for some class in $\text{TxtBc}$.
- From $\text{NonUTxtEx} = \text{TxtEx}$, U-shaped learning is not needed for $\text{TxtEx}$ learning, i.e., for learning ONE successful grammar in limit.
- From $\text{NonUTxtFex}_* \rightarrow \text{TxtFex}_2$, U-shaped learning is needed for some class in $\text{TxtFex}_2$ even if allow $*$ grammars in limit but, from $\text{TxtFex}_2 \rightarrow \text{NonUTxtBc}$, is not needed if allow infinitely many grammars in limit.

- From the reasoning after the prior frame’s diagram, exists $\mathcal{L} \in (\text{TxtFex}_3 - (\text{NonUTxtFex}_* \cup \text{NonUTxtBc}))$; in particular, U-shaped learning IS needed for this $\mathcal{L} \in \text{TxtFex}_3$ — even if allow infinitely many grammars in limit!

Question: Does the class of tasks humans must learn to be competitive in the genetic marketplace, like this latter $\mathcal{L}$, necessitate U-shaped learning?
Main Results and A Question

- **Main Results:**
  - From **NonUTxtBc → TxtBc**, U-shaped learning is needed for some class in **TxtBc**.
  - From **NonUTxtEx = TxtEx**, U-shaped learning is not needed for **TxtEx** learning, i.e., for learning ONE successful grammar in limit.
  - From **NonUTxtFex* → TxtFex_2**, U-shaped learning is needed for some class in **TxtFex_2** even if allow * grammars in limit but, from **TxtFex_2 → NonUTxtBc**, is not needed if allow infinitely many grammars in limit.
  - From the reasoning after the prior frame’s diagram, exists \( \mathcal{L} \in (TxtFex_3 - (NonUTxtFex* \cup NonUTxtBc)) \); in particular, U-shaped learning IS needed for this \( \mathcal{L} \in TxtFex_3 \) — even if allow infinitely many grammars in limit!

- **Question:** Does the class of tasks humans must learn to be competitive in the genetic marketplace, like this latter \( \mathcal{L} \), necessitate U-shaped learning?
Machine Self-Reference

A SELF-REFERENTIAL ROBOT:

Know thyself.
— Greek proverb

Problem: Discover mathematically why above might be good advice.

Initial Results:
No class of (recursive or non-recursive) denotational control structures characterizes the presence of arbitrarily usable self-referential programs in a universal programming language.

A coded-pipelining control structure epitomizes the complement of the latter.
Machine Self-Reference

- **A SELF-REFERENTIAL ROBOT:**

- **Know thyself.**
  — Greek proverb

- **Problem:** Discover mathematically why above might be good advice.

- **Initial Results:**
  - No class of (recursive or non-recursive) denotational control structures characterizes the presence of arbitrarily usable self-referential programs in a universal programming language.
  - A coded-pipelining control structure epitomizes the complement of the latter.
A SELF-REFERENTIAL ROBOT:

Know thyself.
—— Greek proverb

Problem: Discover mathematically why above might be good advice.

Initial Results:
- No class of (recursive or non-recursive) denotational control structures characterizes the presence of arbitrarily usable self-referential programs in a universal programming language.
- A coded-pipelining control structure epitomizes the complement of the latter.
A SELF-REFERENTIAL ROBOT:

Know thyself.
— Greek proverb

Problem: Discover mathematically why above might be good advice.

Initial Results:
- No class of (recursive or non-recursive) denotational control structures characterizes the presence of arbitrarily usable self-referential programs in a universal programming language.
- A coded-pipelining control structure epitomizes the complement of the latter.
Machine Self-Reference

- A SELF-REFERENTIAL ROBOT:

- Know thyself.
  — Greek proverb

- Problem: Discover mathematically why above might be good advice.

- Initial Results:
  No class of (recursive or non-recursive) denotational control structures characterizes the presence of arbitrarily usable self-referential programs in a universal programming language.

  A coded-pipelining control structure epitomizes the complement of the latter.
A SELF-REFERENTIAL ROBOT:

Know thyself.
— Greek proverb

Problem: Discover mathematically why above might be good advice.

Initial Results:

No class of (recursive or non-recursive) denotational control structures characterizes the presence of arbitrarily usable self-referential programs in a universal programming language.

A coded-pipelining control structure epitomizes the complement of the latter.
When unlearning helps.  

J. Case.  
Directions for computability theory beyond pure mathematical.  

J. Case.  
Algorithmic scientific inference: Within our computable expected reality.  

Memory-limited U-shaped learning.  

Non U-shaped vacillatory and team learning.  
J. Case and S. Jain.
Connections between inductive inference and machine learning.

J. Case, S. Jain, S. Kaufmann, A. Sharma, and F. Stephan.
Predictive learning models for concept drift.
Special Issue for ALT’98.

J. Case, S. Jain, E. Martin, A. Sharma, and F. Stephan.
Identifying clusters from positive data.

Refinements of inductive inference by Popperian and reliable machines.

J. Case, S. Jain, M. Ott, A. Sharma, and F. Stephan.
Robust learning aided by context.
Special Issue for COLT’98.

J. Case and T. Kötzing.
Solutions to open questions for non-U-shaped learning with memory limitations.
Expanded version invited for and accepted (with slightly new title) for the associated Special Issue of *TCS*, January 2011.
J. Case and T. Kötzing.
Strongly non U-shaped learning results by general techniques.

J. Case and S. Moelius.
U-shaped, iterative, and iterative-with-counter learning.
Special issue for COLT’07.

J. Case and S. Moelius.
Characterizing programming systems allowing program self-reference.
Special Issue for CiE’2007; online version http://dx.doi.org/10.1007/s00224-009-9168-8.

J. Case and S. Moelius.
Independence results for $n$-ary recursion theorems.
Journal version submitted.

J. Case and S. Moelius.

J. Case and S. Moelius.
Program self-reference in constructive Scott subdomains.
http://dx.doi.org/10.1007/s00224-011-9372-1; Special Issue for CiE’09.
J. Case, M. Ott, A. Sharma, and F. Stephan.  
Learning to win process-control games watching game-masters. 

J. Case and C. Smith.  
Comparison of identification criteria for machine inductive inference. 