# A Characterization of the Language Classes Learnable with Correction Queries*

Cristina Tîrnăucă[1] and Satoshi Kobayashi[2]

[1] Research Group on Mathematical Linguistics, Rovira i Virgili University
Pl. Imperial Tàrraco 1, Tarragona 43005, Spain
cristina.bibire@estudiants.urv.cat
[2] Department of Computer Science, University of Electro-Communications
Chofugaoka 1-5-1, Chofu, Tokyo 182-8585, Japan
satoshi@cs.uec.ac.jp

**Abstract.** Formal language learning models have been widely investigated in the last four decades. But it was not until recently that the model of learning from corrections was introduced. The aim of this paper is to make a further step towards the understanding of the classes of languages learnable with correction queries. We characterize these classes in terms of triples of definite finite tell-tales. This result allowed us to show that learning with correction queries is strictly more powerful than learning with membership queries, but weaker than the model of learning in the limit from positive data.

**Keywords:** correction query, query learning, Gold-style learning.

## 1 Introduction

The field of learning formal languages was practically introduced by E.M. Gold [1] in 1967, in an attempt to construct a precise model for the notion of "being able to speak a language". Gold imagined language learning as an infinite process in which the learner has access to a growing sequence of examples (*learning from text*), or both positive and negative information (*learning from informant*), and is supposed to make guesses. At some point his conjecture should be the target language and he should never change his mind afterwards.

In the same paper Gold also introduces the notion of *finite identification* (from text and informant). The main difference between this model and *learning in the limit model* is that the learner has to stop the presentation of information at some finite time when he "feels" that he has received enough, and state the identity of the target language.

In [2] D. Angluin gives several necessary and sufficient conditions for a class of languages to be learnable in the limit from positive data. Twelve years later, Y. Mukouchi [3] describes the class of languages finitely identifiable from text

* This work was possible thanks to the FPU Fellowship AP2004-6968 from the Spanish Ministry of Education and Science.

J.-Y. Cai, S.B. Cooper, and H. Zhu (Eds.): TAMC 2007, LNCS 4484, pp. 398–407, 2007.

(informant) in terms of definite finite tell-tales (pairs of definite finite tell-tales, respectively).

All the models mentioned so far are also known in the literature as Gold-style learning. A totally different language learning model is the *query learning* model, introduced by Angluin in 1987 [4]. In this setting the learner has access to a truthfully oracle which is allowed to answer specific kind of queries. In [4] a polynomial time query learning algorithm for the class of minimal complete deterministic finite automata (DFAs) is given, in which the learner can ask membership queries (MQs) and equivalence queries (EQs). There are though other types of possible queries: subset queries, superset queries, etc.

Although these two learning models seem to be quite different at a first glance, S. Lange and S. Zilles showed that in fact there is a strong correlation between them [5]. They proved that the class of languages learnable from MQs only coincides with the class of languages finitely identifiable from an informant, and that learning from EQs is equally powerful as learning in the limit from an informant.

As previously mentioned, the study of formal languages learning has its origins in the desire of a better understanding of how children learn so effortlessly their native language. Still, none of these models accurately describes the process of human language learning. Moreover, even the presence of negative information in the process of children language acquisition is subject to a long and still unsolved debate. Clearly, children are not explicitly provided negative examples (words that are not in the language or ungrammatical sentences). Yet, they are corrected when they make a mistake, and this can be thought of as negative information. Actually, these ideas can be found in Gold's paper [1]. Although he points out that "those working in the field generally agree that most children are rarely informed when they make grammatical errors, and those that are informed take little heed", he suggests that maybe "the child receives negative instances by being corrected in a way we do not recognize".

Motivated by these aspects of human language acquisition, L. Becerra-Bonache, A.H. Dediu and C. Tîrnăucă introduced in [6] a new type of query, namely correction query (CQ). A CQ is a slightly modified type of MQ: instead of a 'yes'/'no' answer, the learner receives a correcting string (given $s$ in $\Sigma^*$, the correcting string of $s$ with respect to the language $L$ is the smallest strings $s'$ such that $ss'$ belongs to $L$, if such string exists, and a special symbol otherwise). The same article presents a polynomial time algorithm which infers minimal complete DFAs using CQs and EQs.

In this paper we characterize the language classes learnable with CQs for which the teacher can be effectively implemented (the answers to CQs are computable) by means of triples of definite finite tell-tales (Section 3). We consider only classes of recursive languages, and neglect time complexity issues. Preliminary notions and results are presented in Section 2. In Section 4, using this characterization, we show some relations between our learning model (learning with CQs) and other well-known learning models (like the model of learning

with MQs, or the model of learning in the limit from positive data). Concluding remarks and future work ideas are presented in Section 5.

## 2 Preliminaries

We assume that the reader is familiar with basic notions from formal language theory. A wealth of further information about this area can be found in [7].

Let $\Sigma$ be a finite alphabet of symbols. By $\Sigma^*$ we denote the set of all finite strings of symbols from $\Sigma$. A *language* is any set of strings over $\Sigma$. The length of a string $w$ is denoted by $|w|$, and the concatenation of two strings $u$ and $v$ by $uv$ or $u \cdot v$. The empty string (i.e., the unique string of length 0) is denoted by $\lambda$. If $w = uv$ for some $u, v \in \Sigma^*$, we say that $u$ is a prefix of $w$ and $v$ is a suffix of $w$. By $Pref(L)$ we denote the set $\{w \in \Sigma^* \mid \exists w' \in \Sigma^* \text{ such that } ww' \in L\}$.

Assume that $\Sigma$ is a totally ordered set and let $\prec_L$ be the lexicographical order on $\Sigma^*$. Then, the *lex-length order* $\prec$ on $\Sigma^*$ is defined by: $u \prec v$ if either $|u| < |v|$, or else $|u| = |v|$ and $u \prec_L v$. In other words, strings are compared first according to length and then lexicographically.

Let $\mathcal{C}$ be a class of recursive languages over $\Sigma^*$. We say that $\mathcal{C}$ is an *indexable class* if there is an effective enumeration $(L_i)_{i\geq 1}$ of all and only the languages in $\mathcal{C}$ such that membership is uniformly decidable, i.e., there is a computable function that, for any $w \in \Sigma^*$ and $i \geq 1$, returns 1 if $w \in L_i$, and 0 otherwise. Such an enumeration will subsequently be called an *indexing* of $\mathcal{C}$.

In the sequel we might say that $\mathcal{C} = (L_i)_{i\geq 1}$ is an indexable class and understand that $\mathcal{C}$ is an indexable class and $(L_i)_{i\geq 1}$ is an indexing of $\mathcal{C}$.

### 2.1 Query Learning

In the query learning model a learner has access to an oracle that truthfully answers queries of a specified kind. A query learner $M$ is an algorithmic device that, depending on the reply on the previous queries, either computes a new query, or returns a hypothesis and halts.

More formally, let $\mathcal{C} = (L_i)_{i\geq 1}$ be an indexable class, let $L \in \mathcal{C}$ and let $M$ be a query learner. We say that $M$ learns $L$ using some type of queries if it eventually halts and its only hypothesis, say $i$, correctly describes $L$, i.e., $L_i = L$. So, $M$ returns its unique and correct guess $i$ after only finitely many queries. Moreover, $M$ learns $\mathcal{C}$ using some type of queries if it learns every $L \in \mathcal{C}$ using queries of the specified type. Below we consider:

*Membership queries.* The input is a string $w$ and the answer is 'yes' or 'no', depending on whether or not $w$ belongs to the target language $L$.

*Correction queries.* The input is a string $w$ and the answer is the smallest string (in lex-length order) $w'$ such that $ww'$ belongs to the target language $L$ if $w \in Pref(L)$, and the special symbol $\theta \notin \Sigma$ otherwise. We denote the correction of a string $w$ with respect to the language $L$ by $C_L(w)$.

*Equivalence queries.* The input is an index $j$ of some language $L_j \in \mathcal{C}$. If $L = L_j$, the answer is 'yes'. Otherwise together with the answer 'no', a counterexample from $(L_j \backslash L) \cup (L \backslash L_j)$ is supplied.

The collections of all indexable classes $\mathcal{C}$ for which there is a query learner $M$ such that $M$ learns $\mathcal{C}$ using membership, correction, and equivalence queries are denoted by $MemQ$, $CorQ$ and $EquQ$, respectively.

In this paper we focus on classes of languages for which $Pref(L_i)$ is recursive for all $i \geq 1$. More precisely, we consider indexable classes $\mathcal{C}$ which have the following property $(A)$: there exists a recursive function $f : \mathbb{N}_+ \times \Sigma^* \to \Sigma^* \cup \{\theta\}$ such that $f(i, w) = v$ if and only if $C_{L_i}(w) = v$ for any $w$ in $\Sigma^*$ and $L_i$ in $\mathcal{C}$.

For this purpose, we denote by $CorQ^{(A)}$ the collection of classes of languages in $CorQ$ for which condition $(A)$ is satisfied. Similarly, $MemQ^{(A)}$ is defined. Clearly, for the language classes in $CorQ^{(A)}$ the answers to the correction queries can be effectively computed. That is why in this case we speak about a teacher instead of an oracle.

### 2.2 Gold-Style Learning

In order to present the Gold-style learning models we need some further notions, briefly explained below (for details, see [1,2,8]).

Let $L$ be a non-empty language. A *text for* $L$ is an infinite sequence $\sigma = w_1, w_1, w_3, \ldots$ such that $\{w_i \mid i \geq 1\} = L$. An *informant for* $L$ is an infinite sequence $\sigma = (w_1, b_1), (w_2, b_2), (w_3, b_3), \ldots$ with $b_i \in \{0, 1\}$, $\{w_i \mid i \geq 1 \text{ and } b_i = 1\} = L$, and $\{w_i \mid i \geq 1 \text{ and } b_i = 0\} = \Sigma^* \backslash L$.

Let $\mathcal{C} = (L_i)_{i \geq 1}$ be an indexable class. An inductive inference machine (IIM) is an algorithmic device that reads longer and longer initial segments $\sigma$ of a text (informant) and outputs numbers as its hypotheses. An IIM returning some $i$ is construed to hypothesize the language $L_i$. Given a text (an informant) $\sigma$ for a language $L \in \mathcal{C}$, $M$ identifies $L$ from $\sigma$ if the sequence of hypotheses output by $M$, when fed $\sigma$, stabilizes on a number $i$ (i.e., past some point $M$ always outputs the hypothesis $i$) with $L_i = L$. We say that $M$ identifies $\mathcal{C}$ from text (informant) if it identifies every $L \in \mathcal{C}$ from every corresponding text (informant).

A slightly modified version is the so called model of conservative learning (see [9,10] for more details). A conservative IIM is only allowed to change its mind in case its actual guess contradicts the data seen so far.

As above, *LimTxt* (*LimInf*) denotes the collection of all indexable classes $\mathcal{C}$ for which there is an IIM $M$ such that $M$ identifies $\mathcal{C}$ from text (informant). One can similarly define *ConsvTxt* and *ConsvInf*, for which the inference machines should be conservative IIMs.

Although an IIM is allowed to change its mind finitely many times before returning its final and correct hypothesis, in general it is not decidable whether or not it has already output its final hypothesis. In case that for a given indexable class $\mathcal{C}$, there exists an IIM $M$ such that given any language $L \in \mathcal{C}$ and any text (or informant) for $L$, the first hypothesis $i$ output by $M$ is already correct (i.e., $L_i = L$), we say that $M$ finitely identifies $\mathcal{C}$ (see [1]). The corresponding models *FinTxt* and *FinInf* are defined as above.

In the sequel we present some characterizations for the classes *FinInf* and *ConsvTxt* in terms of pairs of definite finite tell-tales and finite tell-tales, respectively. Let $\mathcal{C} = (L_i)_{i \geq 1}$ be an indexable class.

**Definition 1 (Angluin, [2]).** *A set $T_i$ is a finite tell-tale of $L_i$ if*
*(1) $T_i$ is a finite subset of $L_i$, and*
*(2) for all $j \geq 1$, if $T_i \subseteq L_j$ then $L_j$ is not a proper subset of $L_i$.*

**Theorem 1 (Lange and Zeugmann, [11]).** *An indexable class $\mathcal{C} = (L_i)_{i\geq 1}$ belongs to ConsvTxt if and only if a finite tell-tale of $L_i$ is uniformly computable for any index $i$, that is, there exists an effective procedure which on any input $i \geq 1$ enumerates a finite tell-tale of $L_i$ and halts.*

**Definition 2 (Mukouchi, [3]).** *A language $L$ is consistent with a pair of sets $\langle T, F\rangle$ if $T \subseteq L$ and $F \subseteq \Sigma^* \backslash L$. The pair $\langle T, F\rangle$ is said to be a pair of definite finite tell-tales of $L_i$ if:*
*(1) $T_i$ is a finite subset of $L_i$, $F_i$ is a finite subset of $\Sigma^* \backslash L_i$, and*
*(2) for all $j \geq 1$, if $L_j$ is consistent with the pair $\langle T, F\rangle$, then $L_j = L_i$.*

**Theorem 2 (Mukouchi, [3]).** *An indexable class $\mathcal{C} = (L_i)_{i\geq 1}$ belongs to FinInf if and only if a pair of definite finite tell-tales of $L_i$ is uniformly computable for any index $i$.*

Moreover, there is a strong relation between query learning models and Gold-style learning models. The following strict hierarchy can be found in [5]: $FinTxt \subset FinInf = MemQ \subset ConsvTxt \subset LimTxt \subset LimInf = EquQ$.

## 3 Characterization of the Class $CorQ^{(A)}$

In this section we show that an indexable class with property $(A)$ is learnable from CQs if and only if each language of that class is uniquely characterized by a triple of finite sets. For this, we need some further definitions and notations.

We say that a language $L$ is *consistent with a triple of sets* $\langle T, F, U\rangle$ if $T \subseteq L$, $F \subseteq \Sigma^* \backslash L$ and $U \subseteq \Sigma^* \backslash Pref(L)$.

The triple $\langle T_i, F_i, U_i\rangle$ is a *triple of definite finite tell-tales of* $L_i$ w.r.t. $\mathcal{C} = (L_i)_{i\geq 1}$ if :
(1) $T_i$, $F_i$ and $U_i$ are finite,
(2) $L_i$ is consistent with $\langle T_i, F_i, U_i\rangle$, and
(3) for all $j \geq 1$, if $L_j$ is consistent with $\langle T_i, F_i, U_i\rangle$, then $L_i = L_j$.

**Theorem 3.** *Let $\mathcal{C} = (L_i)_{1\geq 1}$ be an indexable class with property $(A)$. Then $\mathcal{C}$ belongs to CorQ if and only if a triple of definite finite tell-tales of $L_i$ is uniformly computable for any index $i$.*

The theorem is a direct consequence of the following two propositions.

**Proposition 1 (Sufficient condition).** *Let $\mathcal{C} = (L_i)_{1\geq 1}$ be an indexable class. If a triple of definite finite tell-tales of $L_i$ is uniformly computable for any index $i$, then $\mathcal{C}$ is in CorQ.*

*Proof.* Let $\mathcal{C} = (L_i)_{i\geq 1}$ be an indexable class for which a triple of definite finite tell-tales $\langle T_i, F_i, U_i\rangle$ is uniformly computable for any index $i$, and let $w_1, w_2, \ldots$ be the lex-length enumeration of all words in $\Sigma^*$. If $L$ is the target language, then the following query learning algorithm identifies $L$ using CQs.

**Algorithm 1.** A correction query algorithm for the language $L$ in $\mathcal{C}$

```
 1: T := ∅, F := ∅, U := ∅, j := 1
 2: while TRUE do
 3:    get from the oracle the answer to C_L(w_j)
 4:    if (C_L(w_j) = θ) then
 5:       U := U ∪ {w_j}
 6:       F := F ∪ {w_j}
 7:    else
 8:       T := T ∪ {w_j · C_L(w_j)}
 9:       if C_L(w_j) ≠ λ then
10:          F := F ∪ {w_j}
11:       end if
12:    end if
13:    for i := 1 to j do
14:       if (T_i ⊆ T, F_i ⊆ F and U_i ⊆ U) then
15:          output i and halt
16:       end if
17:    end for
18:    j := j + 1
19: end while
```

It is not very difficult to see that if our algorithm outputs an hypothesis, then it is the correct one. Since we constructed $T, F$ and $U$ such that $T \subseteq L, F \subseteq \Sigma^* \backslash L$ and $U \subseteq \Sigma^* \backslash Pref(L)$, it is clear that as soon as we have $T_i \subseteq T$, $F_i \subseteq F$ and $U_i \subseteq U$, the target language $L$ will be consistent with the triple $\langle T_i, F_i, U_i \rangle$, and hence the algorithm outputs $i$ such that $L_i = L$.

Now, let us prove that after asking a finite number of queries, the sets $T$, $F$ and $U$ will be large enough to include $T_i$, $F_i$ and $U_i$, respectively, where $i$ is the smallest index such that $L_i = L$. Let $k_1, k_2, k_3$ and $k$ be such that $k_1 = \max\{j \mid w_j \in T_i\}$, $k_2 = \max\{j \mid w_j \in F_i\}$, $k_3 = \max\{j \mid w_j \in U_i\}$ and $k = \max\{k_1, k_2, k_3, i\}$.

Consider the sets $T, F, U$ constructed after receiving the corrections for the strings $w_1, s_2, \ldots, s_k$.

1. If $w \in T_i$, then $w \preceq w_k$ and $C_L(w) = \lambda$. Hence, $w \in T$.
2. If $w \in U_i$, then $w \preceq w_k$ and $C_L(w) = \theta$. Hence, $w \in U$.
3. If $w \in F_i$, then $w \preceq w_k$ and $C_L(w) \neq \lambda$. We distinguish two cases. Either $C_L(w) \in \Sigma^+$ and then $w$ is added to $F$ at line 10 of the algorithm, or $C_L(w) = \theta$ and $w$ is added to $F$ at line 6 of the algorithm. In both of the cases, $w \in F$.

We have seen that after reading corrections of at most $k$ strings, $T_i \subseteq T$, $F_i \subseteq F$ and $U_i \subseteq U$, and since $i$ is smaller than or equal to $k$, the algorithm outputs the (correct) hypothesis $i$. □

**Proposition 2 (Necessary condition).** *If $\mathcal{C} = (L_i)_{1 \geq 1}$ is in $CorQ^{(A)}$ then a triple of definite finite tell-tales of $L_i$ is uniformly computable for any index $i$.*

*Proof.* Let $\mathcal{C} = (L_i)_{i \geq 1}$ be an indexable class in $CorQ^{(A)}$, and take $M$ to be a query learning algorithm which learns $\mathcal{C}$ using CQs. The following procedure computes a triple of definite finite tell-tales of $L_i$ for any $i \geq 1$.

**Algorithm 2.** Computing a triple of definite finite tell-tales

1: Input: the target language $L_i$
2: run $M$ on $L_i$, and collect the sequence of queries and answers in $QA_i$
3: $T_i := \{wv \mid (w,v) \in QA_i, v \neq \theta\}$
4: $F_i := \{wv' \mid (w,v) \in QA_i, v \neq \theta, v' \prec v\}$
5: $U_i := \{w \mid (w,\theta) \in QA_i\}$
6: output $\langle T_i, F_i, U_i\rangle$ and halt.

Clearly, $T_i$, $F_i$ and $U_i$ are all finite. We show that $T_i \subseteq L_i$, $F_i \subseteq \Sigma^* \backslash L_i$ and $U_i \subseteq \Sigma^* \backslash Pref(L_i)$. If $u \in T_i$, then there exist $w, v$ in $\Sigma^*$ such that $u = wv$ and $v = C_{L_i}(w)$. Hence, $u = wv \in L_i$. If $u \in F_i$, then there exist $w, v, v'$ in $\Sigma^*$ such that $v' \prec v$, $u = wv'$ and $v = C_{L_i}(w)$. Hence, $u = wv' \notin L_i$. If $u \in U_i$, then $C_{L_i}(u) = \theta$, and hence $u \in \Sigma^* \backslash Pref(L_i)$.

Let us now take $j$ such that $L_j$ is consistent with the triple $\langle T_i, F_i, U_i\rangle$. We compute $C_{L_j}(w)$ for each pair $(w,v)$ in $QA_i$. If $v = \theta$, then $w \in U_i$. But $U_i \subseteq \Sigma^* \backslash Pref(L_j)$ implies $w \notin Pref(L_j)$, and hence $C_{L_j}(w) = \theta$. If $v \in \Sigma^* \backslash \{\theta\}$, then $wv \in T_i$ and $wv' \in F_i$ for all $v' \prec v$. But $T_i \subseteq L_j$ and $F_i \subseteq \Sigma^* \backslash L_j$ implies $wv \in L_j$ and $wv' \notin L_j$ for all $v' \prec v$. Hence, $C_{L_j}(w) = v$.

We have shown that for all $(w,v) \in QA_i$, $C_{L_j}(w) = v = C_{L_i}(w)$. Since the algorithm $M$ is assumed to identify a unique language from the class $\mathcal{C}$, we obtain $L_i = L_j$. This makes $\langle T_i, F_i, U_i\rangle$ a triple of definite finite tell-tales of $L_i$. □

## 4 Relations to Other Learning Models

Using the results presented in the previous section, we show the relations between correction query learning models and other learning models.

### 4.1 A Model Included in *CorQ*

Let $\mathcal{C} = (L_i)_{i \geq 1}$ be an indexable class. We have the following theorem.

**Theorem 4.** *If $\mathcal{C}$ is in FinInf, then $\mathcal{C}$ is in CorQ.*

*Proof.* Assume that $\mathcal{C}$ is in *FinInf*. Then cf. Theorem 2, a pair of definite finite tell-tales $\langle T_i, F_i\rangle$ of $L_i$ is uniformly computable for any index $i$. We show that $\langle T_i, F_i, \emptyset\rangle$ is a triple of definite finite tell-tales for $L_i$. Clearly, $T_i$ is a finite subset of $L_i$, $F_i$ is a finite subset of $\Sigma^* \backslash L_i$ and the empty set is a finite subset of $\Sigma^* \backslash Pref(L_i)$. Let us now take $j$ such that $L_j$ is consistent with the triple $\langle T_i, F_i, \emptyset\rangle$. Because $\langle T_i, F_i\rangle$ is a pair of definite finite tell-tales for $L_i$, $T_i \subseteq L_j$ and $F_i \subseteq \Sigma^* \backslash L_j$, we obtain $L_j = L_i$, and hence $\langle T_i, F_i, \emptyset\rangle$ is a triple of definite finite tell-tales for $L_i$. Using Proposition 1, we immediately get that $\mathcal{C}$ is in *CorQ*. □

Let us now show that the inclusion is strict. Take $K_1, K_2, K_3, \ldots$ to be the collection of all finite non-empty sets of positive integers (indexable somehow). Take $\Sigma = \{a\}$, and define $L_i = \{a^n \mid n \in K_i\}$ for all $i \geq 1$. Clearly, $\mathcal{C}^{CorQ}_{FinInf} = (L_i)_{i \geq 1}$ is an indexable class.

**Lemma 1.** *$\mathcal{C}^{CorQ}_{FinInf}$ is in CorQ.*

*Proof.* We show that $\langle T_i, F_i, U_i\rangle$ is a triple of definite finite tell-tales of $L_i$ for any index $i$, where $T_i = L_i$, $l = \max\{n \mid n \in K_i\}$, $F_i = \{a^n \mid n \in \{1,\ldots,l\}\backslash K_i\}$ and $U_i = \{a^{l+1}\}$. Indeed, it is easy to see that $T_i$, $F_i$, $U_i$ are finite, and that $T_i \subseteq L_i$, $F_i \subseteq \Sigma^*\backslash L_i$ and $U_i \subseteq \Sigma^*\backslash Pref(L_i)$. Let us take $j$ such that $L_j$ is consistent with the triple $\langle T_i, F_i, U_i\rangle$. Then, $F_i \subseteq \Sigma^*\backslash L_j$ implies $(\{1,\ldots,l\}\backslash K_i) \cap K_j = \emptyset$, and $U_i \subseteq \Sigma^*\backslash Pref(L_j)$ implies $K_j \subseteq \{1,\ldots,l\}$. Putting together these last two results we obtain $K_j \subseteq K_i$, and hence $L_j \subseteq L_i$. But $T_i \subseteq L_j$ implies $L_i \subseteq L_j$. So, $L_j = L_i$ which concludes the proof. □

**Lemma 2.** *$\mathcal{C}^{CorQ}_{FinInf}$ is not in FinInf.*

*Proof.* Now, assume that $\mathcal{C}^{CorQ}_{FinInf}$ is in *FinInf*. Cf. Theorem 2, this implies that a pair of definite finite tell-tales $\langle T_i, F_i\rangle$ of $L_i$ is uniformly computable for any index $i$. Let us fix $i$, take $l = \max\{i \mid a^i \in F_i\}$, and set $j$ to be the index for which $K_j = K_i \cup \{l+1\}$. Then, $L_j$ is also consistent with the pair $\langle T_i, F_i\rangle$ since $T_i \subseteq L_i \subset L_j$ and $F_i \subseteq \Sigma^*\backslash L_j$ ($F_i \subseteq \Sigma^*\backslash L_i$ and $a^{l+1} \notin F_i$), and hence $L_j = L_i$. We reach a contradiction since $a^{l+1} \in L_j\backslash L_i$. □

This result can be extended to any alphabet $\Sigma = \{a_1, a_2, \ldots, a_n\}$ if we set $L_i$ to be $\{a_1a_2\ldots a_{n-1}a_n^j \mid j \in K_i\}$ for any index $i$.

As a direct consequence, we obtain that the class *MemQ* is strictly included in *CorQ*. So, CQs are strictly more powerful than MQs, and they cannot be simulated by a finite number of MQs.

### 4.2 A Model Which Includes $CorQ^{(A)}$

Let $\mathcal{C} = (L_i)_{i\geq 1}$ be an indexable class. We have the following theorem.

**Theorem 5.** *If $\mathcal{C}$ is in $CorQ^{(A)}$, then $\mathcal{C}$ is in ConsvTxt.*

*Proof.* If $\mathcal{C} = (L_i)_{i\geq 1}$ is in $CorQ^{(A)}$ then, cf. Proposition 2, a triple of definite finite tell-tales $\langle T_i, F_i, U_i\rangle$ of $L_i$ is uniformly computable for any index $i$.

We show that $T_i$ is a finite tell-tale for $L_i$. Clearly, $T_i$ is a finite subset of $L_i$. Let us now take $j$ such that $T_i \subseteq L_j$. We need to prove that $L_j$ is not a proper subset of $L_i$. Assume by contrary that it is. Then, $L_j \subset L_i$ implies $Pref(L_j) \subseteq Pref(L_i)$, and hence $\Sigma^*\backslash Pref(L_j) \supseteq \Sigma^*\backslash Pref(L_i)$. Keeping in mind that $U_i \subseteq \Sigma^*\backslash Pref(L_i)$ we obtain that $U_i \subseteq \Sigma^*\backslash Pref(L_j)$. Moreover, $F_i \subseteq \Sigma^*\backslash L_i$ and $\Sigma^*\backslash L_j \supseteq \Sigma^*\backslash L_i$ imply $F_i \subseteq \Sigma^*\backslash L_j$. Since $L_j$ is consistent with the triple $\langle T_i, F_i, U_i\rangle$, we have $L_i = L_j$ which contradicts our assumption. So given any index $i$, a finite tell-tale of $L_i$ is uniformly computable. □

Let us now show that the inclusion is strict. For this, we denote by $I(n)$ the set of all positive integral multiples of $n$. Let the collection of all finite non-empty sets of prime positive integers be $P_1, P_2, P_3, \ldots$ indexable, for example, in order of increasing $\prod_{p\in P_i} p$. Then, take $\Sigma = \{a\}$, $R_i = \cup_{p\in P_i} I(p)$ and $L_i = \{a^n \mid n \in R_i\}$. Clearly, $\mathcal{C}^{ConsvTxt}_{CorQ} = (L_i)_{i\geq 1}$ is an indexable class.

**Lemma 3.** $\mathcal{C}_{CorQ}^{ConsvTxt}$ *is in* $ConsvTxt$.

*Proof.* Let us first notice that $T_i = \{a^p \mid p \in P_i\}$ is a finite tell-tale for $L_i$. Clearly, $T_i$ is a finite subset of $L_i$. If we take $j$ such that $L_j \supseteq T_i$, we have $R_j \supseteq P_i$, and furthermore $R_j \supseteq R_i$ and $L_j \supseteq L_i$. Hence, $L_j$ is not a proper subset of $L_i$. Moreover, $P_i$ is uniformly computable for any index $i$, and so is $T_i$. □

**Lemma 4.** $\mathcal{C}_{CorQ}^{ConsvTxt}$ *is not in* $CorQ^{(A)}$.

*Proof.* Assume by contrary that $\mathcal{C}_{CorQ}^{ConsvTxt}$ is in $CorQ^{(A)}$. Cf. Proposition 2, a triple of definite finite tell-tales $\langle T_i, F_i, U_i\rangle$ of $L_i$ is uniformly computable, for any index $i$.

We introduce the following notation: $Num(S) = \{|w| \mid w \in S\}$ for any set $S \subseteq \Sigma^*$. Then $T_i \subseteq L_i$ is equivalent to $Num(T_i) \subseteq R_i$, and $F_i \subseteq \Sigma^* \backslash L_i$ is equivalent to $Num(F_i) \cap R_i = \emptyset$. Finally, $U_i \subseteq \Sigma^* \backslash Pref(L_i)$ implies $U_i = \emptyset$.

Let us now choose a prime number $p$ such that $I(p) \cap Num(F_i) = \emptyset$ and $p \notin P_i$, and take $j$ such that $P_j = P_i \cup \{p\}$. Clearly, $L_i \subset L_j$. We show that $L_j$ is consistent with $\langle T_i, F_i, U_i\rangle$.

Indeed, $T_i \subseteq L_j$ because $T_i \subseteq L_i$ and $L_i \subset L_j$. Also, $Num(F_i) \cap R_j = \emptyset$ because $Num(F_i) \cap R_i = \emptyset$, $Num(F_i) \cap I(p) = \emptyset$ and $R_j = R_i \cup I(p)$. Hence, $F_i \subseteq \Sigma^* \backslash L_j$. The empty set is trivially included in any set, and hence $U_i \subseteq \Sigma^* \backslash Pref(L_j)$.

We found an index $j$ such that $L_j$ is consistent with $\langle T_i, F_i, U_i\rangle$ and $L_j \neq L_i$ which is a contradiction. □

## 5 Concluding Remarks

Learning formal languages has been a preoccupation of many researchers during time. With every new learning model introduced, many research possibilities were created. Since very recently a new model in the query learning theory has been proposed, namely learning with CQs, we considered that it is worth investigating the particularities of the classes of languages learnable within this setting.

We showed that there exists a method of characterizing these classes using some finite triples, called triples of definite finite tell-tales. With the help of this characterization, we managed to position the class $CorQ$ in the hierarchy formed by other well-known learning models (both Gold-style learning and query learning models).

As we already mentioned, our work was focused on the type of correction introduced in [6]. In the future we would like to consider other types of corrections and to answer to the following question: how is $CorQ$ influenced by the type of correction used? What about combining CQs and a limited number of EQs, or CQs and positive examples? What happens if we restrict to polynomial time learning? Can we construct a language class which is in $CorQ$ and not in $CorQ^{(A)}$? We believe that these are several question which deserve further investigation.

## References


1. Gold, E.M.: Language identification in the limit. Information and Control **10** (1967) 447–474
2. Angluin, D.: Inductive inference of formal languages from positive data. Information and Control **45** (1980) 117–135
3. Mukouchi, Y.: Characterization of finite identification. In Jantke, K.P., ed.: Proc. 3rd International Workshop on Analogical and Inductive Inference (AII '92). Volume 642 of Lecture Notes in Computer Science., London, UK, Springer-Verlag (1992) 260–267
4. Angluin, D.: Learning regular sets from queries and counterexamples. Information and Computation **75** (1987) 87–106
5. Lange, S., Zilles, S.: Formal language identification: query learning vs. Gold-style learning. Information Processing Letters **91** (2004) 285–292
6. Beccera-Bonache, L., Dediu, A.H., Tîrnăucă, C.: Learning DFA from correction and equivalence queries. In Sakaibara, Y., Kobayashi, S., Sato, K., Nishino, T., Tomita, E., eds.: Grammatical Inference: Algorithms and Applications, Proceedings of ICGI '06. Volume 4201 of Lecture Notes in Artificial Intelligence., Berlin, Heidelberg, Springer-Verlag (2006) 281–292
7. Martín-Vide, C., Mitrana, V., Păun, G., eds.: Formal Languages and Applications. Studies in Fuzzyness and Soft Computing 148. Springer-Verlag, Berlin, Heidelberg (2004)
8. Zeugmann, T., Lange, S.: A guided tour across the boundaries of learning recursive languages. In Jantke, K.P., Lange, S., eds.: Algorithmic Learning for Knowledge-Based Systems, GOSLER Final Report. Volume 961 of Lecture Notes in Computer Science., London, UK, Springer-Verlag (1995) 190–258
9. Zeugmann, T., Lange, S., Kapur, S.: Characterizations of monotonic and dual monotonic language learning. Information and Computation **120** (1995) 155–173
10. Zeugmann, T.: Inductive inference and language learning. In Cai, J., Cooper, S.B., Li, A., eds.: Proc. 3rd International Conference on Theory and Applications of Models of Computation (TAMC '06). Volume 3959 of Lecture Notes in Computer Science., Berlin, Heidelberg, Springer-Verlag (2006) 464–473
11. Lange, S., Zeugmann, T.: Types of monotonic language learning and their characterization. In: Proc. 5th Annual Workshop on Computational Learning Theory (COLT '92), New York, ACM Press (1992) 377–390