REVERSIBLE AUTOMATA AND INDUCTION OF THE ENGLISH AUXILIARY SYSTEM

Samuel F. Pilato
Robert C. Berwick
MIT Artificial Intelligence Laboratory
545 Technology Square
Cambridge, MA 02139, USA

ABSTRACT

In this paper we apply some recent work of Angluin (1982) to the induction of the English auxiliary verb system. In general, the induction of finite automata is computationally intractable. However, Angluin shows that restricted finite automata, the *k-reversible* automata, can be learned by efficient (polynomial time) algorithms. We present an explicit computer model demonstrating that the English auxiliary verb system can in fact be learned as a 1-reversible automaton, and hence in a computationally feasible amount of time. The entire system can be acquired by looking at only half the possible auxiliary verb sequences, and the pattern of generalization seems compatible with what is known about human acquisition of auxiliaries. We conclude that certain linguistic subsystems may well be learnable by inductive inference methods of this kind, and suggest an extension to context-free languages.

INTRODUCTION

Formal inductive inference methods have rarely been applied to actual natural language systems. Linguists generally suppose that languages are easy to learn because grammars are highly constrained; no "general purpose" inductive inference methods are required. This assumption has generally led to fruitful insights on the nature of grammars. Yet it remains to determine whether *all* of a language is learned in a grammar-specific manner. In this paper we show how to successfully apply one computationally efficient inductive inference algorithm to the acquisition of a domain of English syntax. Our results suggest that particular language subsystems can be learned by general induction procedures, given certain general constraints.

The problem is that these methods are in general computationally intractable. Even for regular languages induction can be exponentially difficult (Gold, 1978). This suggests that there may be general constraints on the design of certain linguistic subsystems to make them easy to learn by general inductive inference methods. We propose the constraint of *k-reversibility* as one such restriction. This constraint guarantees polynomial time inference (Angluin, 1982). In the remainder of this paper, we also show, by an explicit computer model, that the English auxiliary verb system meets this constraint, and so is easily inferred from a corpus. The theory gives one precise characterization of just where we may expect general inductive inference methods to be of value in language acquisition.

LEARNING *K*-REVERSIBLE LANGUAGES FROM EXAMPLES

The question we address is, If a learner presumes that a natural language domain is systematic in some way, can the learner intelligently infer the complete system from only a subset of sample sentences? Let us develop an example to formally describe what we mean by "systematic in some way," and how such a systematic domain allows the inference of a complete system from examples. If you were told that *Mary bakes cakes*, *John bakes cakes*, and *Mary eats pies* are legal strings in some language, you might guess that *John eats pies* is also in that language. Strings in the language seem to follow a recognizable pattern, so you expect other strings that follow the same pattern to be in the language also.

In this particular case, you are presuming that the to-be-learned language is a zero-reversible regular language. Angluin (1982) has defined and explored the formal properties of reversible regular languages. We here translate some of her formal definitions into less technical terms.

A regular language is any language that can be generated from a formula called a regular expression. For example the strings mentioned above might have come from the language that the following regular expression generates:

(Mary|John) (bakes|eats) [[very* delicious] (cakes|pies)]

A complete natural language is too complex to be generated by some concise regular expression, but some simple subsets of a natural language can fit this kind of pattern.

To formally define when a regular language is reversible, let us first define a prefix as any substring (possibly zero-

Table 1: Example of incremental k-reversible inference for several values of k.

SEQUENCE OF NEW STRINGS PRESENTED	NEW STRINGS INFERRED:		
	$k = 0$	$k = 1$	$k = 2$
Mary bakes cakes	NONE	NONE	NONE
John bakes cakes	NONE	NONE	NONE
Mary eats pies	John eats pies	NONE	NONE
Mary bakes pies	John bakes pies Mary eats cakes John eats cakes	John bakes pies	NONE
Mary bakes	John bakes Mary eats John eats Mary bakes cakes cakes John bakes cakes cakes Mary bakes pies cakes . . . (Mary|John)(bakes|eats)(cakes|pies)*	John bakes	NONE

length) that can be found at the very beginning of some legal string in a language, and a suffix as any substring (again, possibly zero-length) that can be found at the very end of some legal string in a language. In our case the strings are sequences of words, and the language is the set of all legal sentences in our simplified subset of English. Also, in any legal string say that the suffix that immediately follows a prefix is a tail for that prefix. Then a regular language is *zero-reversible* if whenever two prefixes in the language have a tail in common, then the two prefixes have all tails in common.

In the above example, prefixes *Mary* and *John* have the tail *bakes cakes* in common. If we presume that the language these two strings come from is zero-reversible, then *Mary* and *John* must have all tails in common. In particular, the third string shows that *Mary* has *eats pies* as a tail, so *John* must also have *eats pies* as a tail. Our current hypothesis after having seen these three strings is that they come not from the three-string language expressed by (*Mary*|*John*) *bakes cakes* | *Mary eats pies*, which is not zero-reversible, but rather from the four-string language (*Mary*|*John*) (*bakes cakes* | *eats pies*), which is zero-reversible. Notice that we have enlarged the corpus just enough to make the language zero-reversible.

A regular language is *k-reversible*, where k is a non-negative integer, if whenever two prefixes *whose last k words match* have a tail in common, then the two prefixes have all tails in common. A higher value of k gives a more conservative condition for inference. For example, if we presume that the aforementioned strings come from a 1-reversible language, then instead of presuming that whatever *Mary* does *John* does, we would presume only that whatever *Mary bakes*, *John bakes*. In this case the third string fails to yield any inference, but if we were later told that *Mary bakes pies* is in the language, we could infer that *John bakes pies* is also in the language. Further adding the sentence *Mary bakes* would allow 1-reversible inference to also induce *John bakes*, resulting in the seven-string 1-reversible language expressed by (*Mary*|*John*) *bakes* (*cakes*|*pies*) | *Mary eats pies*.

With these examples zero-reversible inference would have generated (*Mary*|*John*) (*bakes*|*eats*) (*cakes*|*pies*)* by now, which overgeneralizes an *optional* direct object into *zero or more* direct objects. On the other hand, two-reversible inference would have inferred no additional strings yet. For a particular language we hope to find a k that is small enough to yield some inference but not so small that we overgeneralize and start inferring strings that are in fact not in the true language we are trying to learn. Table 1 summarizes our examples of k-reversible inference.

AN INFERENCE ALGORITHM

In addition to formally characterizing k-reversible languages, Angluin also developed an algorithm for inferring a k-reversible language from a finite set of positive examples, as well as a method for discovering an appropriate k when negative examples (strings known not to be in the language) are also presented. She also presented an algorithm for determining, given some k-reversible regular language, a minimal set of examples from which the entire language can be induced. We have implemented these procedures on a computer in MACLISP and have applied them to all of the artificial languages in Angluin's paper as well as to all of the natural language examples in this paper.

To describe the inference algorithm, we make use of the fact that every regular language can be associated with a corresponding deterministic finite-state automaton (DFA) which accepts or generates exactly that language.

Given a sample of strings taken from the full corpus, we first generate a prefix-tree automaton which accepts or generates exactly those strings and no others. We now want to infer additional strings so as to induce a k-reversible language, for some chosen k. Let us say that when accepting a string, the last k symbols encountered before arriving at a state is a *k-leader* of that state. Then to generalize the language, we recursively merge any two states where any of the following is true:

- Another state arcs to both states on the same word. (This enforces determinism.)
- Both states have a common k-leader and either
  - both states are accepting states or
  - both states arc to a common state on the same word.

When none of these conditions obtains any longer, the resulting DFA accepts or generates the smallest k-reversible language that includes the original sample of strings. (The term "reversible" is used because a k-reversible DFA is still deterministic with lookahead k when its sets of initial and final states are swapped and all of its arcs are reversed.)

This procedure works incrementally. Each new string may be added to the DFA in prefix-tree fashion and the state-merging algorithm repeated. The resulting language induced is independent of the order of presentation of sample strings.

If an appropriate k is not known *a priori*, but some negative as well as positive examples are presented, then one can try increasing values of k until the induced language contains none of the negative examples.

Though the inference algorithm takes a sample and induces a k-reversible language, it is quite helpful to use Angluin's algorithm for going in the reverse direction: given a k-reversible language we can determine what minimal set of shortest possible examples (a "characteristic" or "covering" sample) will be sufficient for inducing the language. Though the minimal *number* of examples is of course unique, the set of particular strings in the covering sample is not necessarily unique.

INFERENCE OF THE ENGLISH AUXILIARY SYSTEM

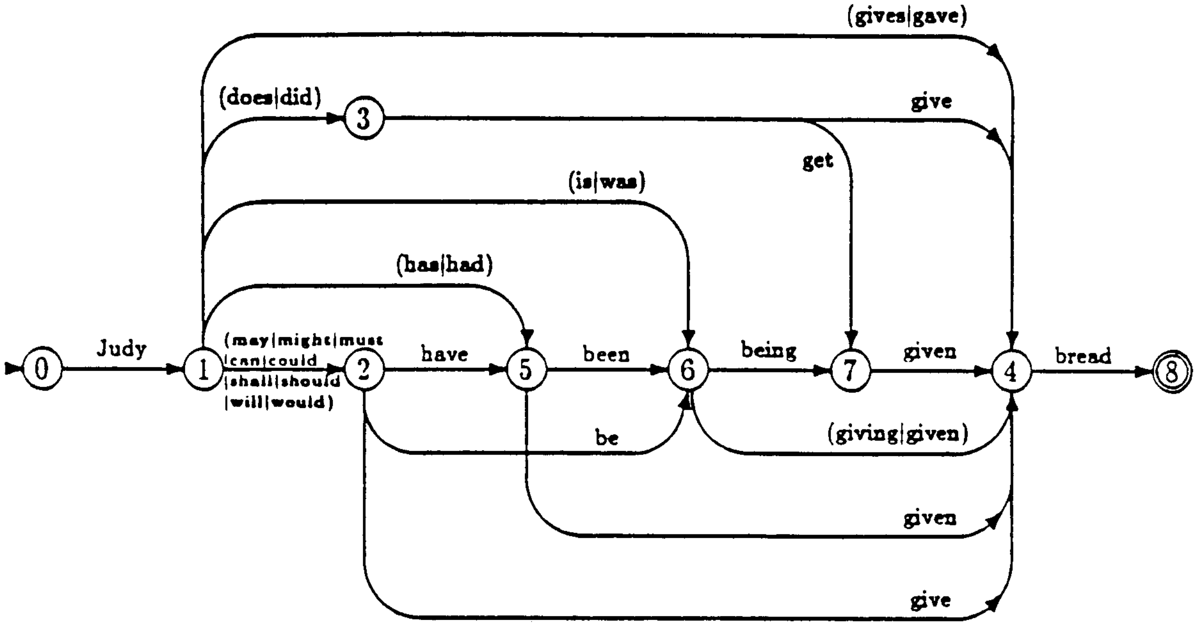
We have chosen to test the English auxiliary system under k-reversible inference because English verb sequences are highly regular, yet they have some degree of complexity and admit to some exceptions. We represent the English auxiliary system as a corpus of 92 variants of a declarative statement in third person singular. The variants cover all standard legal permutations of tense, aspect, and voice, including *do* support and nine modals. We simply use the surface forms, which are strings of words with no additional information such as syntactic category or root-by-inflection breakdown. For instance, the present, simple, active example is *Judy gives bread*. One modal, perfective, passive variant is *Judy would have been given bread*.

We have explored the k-reversible properties of this natural language subsystem in two main steps. First we determined for what values of k the corpus is in fact k-reversible. (Given a finite corpus, we could be sure the language is k-reversible for all k at or above some value.) To do this we treated the full corpus as a set of sample strings and tried successively larger values of k until finding one where k-reversible inference applied to the corpus generates no additional strings. We could then be sure that any k of that value or greater could be used to infer an accurate model of the English auxiliary system without overgeneralizing.

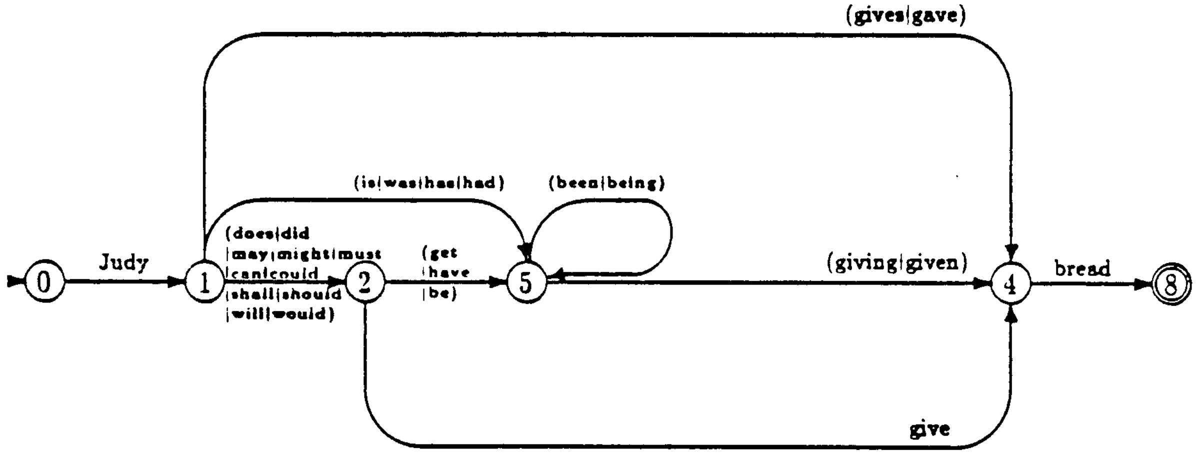
After finding the range of values of k to work with, we were interested in determining which, if any, of those values of k would yield some power to infer the full corpus from a proper subset of examples. To do this we took the DFA which represents the full corpus and computed, for a trial k, a set of sample strings that would be minimally sufficient to induce the full corpus. If any such values of k exist, then we can say that, in a nontrivial way, the English auxiliary system is learnable as a k-reversible language from examples.

We found that the English auxiliary system can be faithfully modeled as a k-reversible regular language for $k \geq 1$. Only zero-reversible inference overgeneralizes the full corpus as well as the active and passive corpora treated as separate languages. For the active corpus, zero-reversible inference groups the forms of *do* with the other modals. The DFAs for the passive and full corpora also contain loops and thereby generate infinite numbers of illegal variants.

Figure 1 compares a correct DFA for the English auxiliary system with an overgeneralized DFA. Both are shown in a minimized, canonical form. The top, correct, automaton can be generated by either minimizing the prefix tree for the full corpus or by minimizing the result of k-reversible inference applied to any sufficiently characteristic set of sample sentences, for any $k \geq 1$. One can read off all 92 variants



THE ENGLISH AUXILIARY SYSTEM



ZERO-REVERSIBLE OVERGENERALIZATION
OF THE ENGLISH AUXILIARY SYSTEM

Figure 1: The top automaton generates the English auxiliary system. Zero-reversible inference merges state 3 with state 2 and merges states 7 and 6 with state 5, resulting in the bottom overgeneralized version.

in the language by taking different paths from initial state to final state. The bottom, overgeneralized, automaton is generated by subjecting the top one to zero-reversible inference.

Does treating the English auxiliary system as a 1-or-more-reversible language yield any inferential power? The English auxiliary system as a 1-reversible language can in fact be inferred from a cover of only 48 examples out of the 92 variants in the corpus. The active corpus treated separately requires 38 examples out of 46 and the passive corpus requires 28 out of 46. Treating the full corpus as a 2-reversible language requires 76 examples, and a 3^+-reversible model cannot infer the corpus from any proper subset whatsoever.

For 1-reversible inference, 45 of the verb sequences of length three or shorter will yield the remaining nine such strings and none longer. Verb sequences of length four or five can be divided into two patterns, *<modal> have been giv(ing;en)* and ... *be(en) being given*. Adding any one (length-four) string from the first pattern will yield the remaining 17 strings of that pattern. Further adding two length-four strings from the awkward second pattern will yield the remaining 18 strings of that pattern, nine of which are of length five. This completes the corpus.

DISCUSSION

The auxiliary system has often been regarded as an acid test for a theory of language acquisition. Given this, we are encouraged that it is in fact learnable via a computationally efficient general method. It is significant that at least in this domain we have found a k (of 1) that is low enough to generate a good amount of inference from examples yet high enough to avoid overgeneralization. Even more conservative 2-reversibility generates a little inference.

This inductive power derives from the systematic sequential structure of the English auxiliary system. In an idealized form (ignoring tense and inflections) the regular expression

[DO | [<modal>] [HAVE] [BE]] [BEpassive] GIVE

generates all English verb sequence patterns in our corpus.

Zero-reversible inference basically attempts to simplify any partial, disjunctive permutation like $(a|b)x|ay$ into an exhaustive, combinatorial permutation like $(a|b)(x|y)$. Since the active corpus (excluding *BE-passive* from the idealized regular expression) in fact has such a simple form except for the *DO* disjunction, zero-reversible inference productively completes the three-place permutation but also destroys the disjunction, by overgeneralizing what patterns can follow both *DO* and *<modal>*. One-reversible inference requires that disjuncts share some final word to be mergeable, so that *DO* cannot merge with any auxiliary triplet, yet the permutation of *<modal> HAVE* by *[BE]* is still productive. Similar considerations obtain in the passive case, as well as for the joint corpus. Table 2 illustrates the trade-off in this case between inferential power and the proper handling of exceptions.

In complex environments, rather than reduce the inferential power by raising k one could instead embed this algorithm within a larger system. For example, a more realistic model of processing English verb sequences would have an external, more linguistically motivated mechanism force the separate treatment of active versus passive forms. Then if, say on considerations of frequency of occurrence, *do* exceptions were externally handled and the infrequent

Table 2: Incremental k-reversible inference of some English auxiliary verb sequences.

SEQUENCE OF NEW	NEW STRINGS INFERRED:		
STRINGS PRESENTED	$k = 0$	$k = 1$	$k = 2$
could give	NONE	NONE	NONE
may give	NONE	NONE	NONE
does give	NONE	NONE	NONE
could have given	may have given does have given	NONE	NONE
may have given	(ALREADY INFERRED)	NONE	NONE
could have been giving	may have been giving does have been giving	may have been giving	NONE

... *BE being* ... cases were similarly excluded from the immature learner, then one could apply the more powerful zero-reversible inference to the remaining active and passive forms without overgeneralizing. In such a case the active system can be induced from 18 examples out of 44 variants and the passive system from 14 out of 22. The entire active system is learnable once examples of each form of each verb and each modal have been seen, plus one example to fix the relative order of *have* vs. *be*, and one example each to fix the order of modal vs. *have* or *be*.

Though a more complex model must ultimately represent a domain like the English auxiliary system, the way k-reversible inference in itself handles a complex territory satisfies some conditions of psychological fidelity. Especially zero-reversibility is a rather simple form of generalization of sequential patterns with which we believe humans readily identify. In general the longer, more complex cases can be inferred from simpler cases. Also, there is a reasonable degree of play in the composition of the covering sample, and the order of presentation does not affect the language learned.

Children evidently never make mistakes on the relative order of auxiliaries, which is consistent with the reversibility model, but they do mistakenly combine *do* with tensed verb forms (Pinker, 1984). Given that the appearance of *do* in declarative sentences is also fairly rare, one might prefer the aforementioned zero-reversible system that handles *do* support as an exception, rather than opt for a 1-reversible inference which is flawless but a slower learner.

The ... *BE being* ... cases are systematically related to the rest, but also have a natural boundary: 1-reversible inference from simpler cases doesn't intrude into that territory, yet only a few such examples allow one to infer the remainder. Very rare sequences like *could have been being given* will be successfully acquired even if they are not seen. This seems consistent with human judgments that such phrasing is awkward but apparently legal.

k-Reversibility is essentially a model of simplicity, not of complexity. As such, it induces not linguistic structure but the substitution classes that linguistic structures typically work with, building these by analogy from examples. In the linguistic structure for which k-reversibility is defined — regular grammars — it functions to induce the classes that fill "slots" in a regular expression, based on the similarity of tail sets. Increasing the value of k is a way of requiring a higher degree of similarity before calling a match. (See Gonzalez and Thomason, 1978, for other approaches to k-tail inference that are not so efficient.)

The same principle can apply to the induction of substitution classes in other linguistic domains including morphological, syntactic, and semantic systems. For a particularly direct example, consider the right-hand sides of context-free rewrite rules. Any subset of such rules having the same left-hand side constitutes a regular language over the set of terminal and nonterminal symbols, and is therefore a candidate for induction. One might thus infer new rewrite rules from the pattern of existing ones, thereby not only concluding that words are members of certain simple syntactic classes, but also simplifying a disjunctive set of rules into a more concise set that exhibits systematic properties. Berwick's *Lparsifal* system (1982) is an example of this kind of extension.

We believe that k-reversibility illustrates a psychologically plausible pattern induction process for natural language learning that in its simplest form has an efficient computational algorithm associated with it. The basic principle behind k-reversible inference shows some promise as a flexible tool within more complex models of language acquisition. It is encouraging that, at least in a simple case, computational linguistic models can suggest formal learnability constraints that are natural enough to be useful in the learning of human languages.

ACKNOWLEDGMENTS

This paper describes research done at the Artificial Intelligence Laboratory of the Massachusetts Institute of Technology. Support for the laboratory's artificial intelligence research is provided in part by the Advanced Research Projects Agency of the Department of Defense under the Office of Naval Research Contract N00014-80-C-0505.

REFERENCES

Angluin, D., "Inference of reversible languages," *Journal of the Association for Computing Machinery*, 29(3), 741–765, 1982.

Berwick, R., *Locality Principles and the Acquisition of Syntactic Knowledge*, PhD, MIT Department of Electrical Engineering and Computer Science, 1982.

Gold, E., "Complexity of Automaton Identification from Given Data," *Information and Control*, 37, 1978.

Gonzalez, R., and Thomason, M., *Syntactic Pattern Recognition*, Reading, MA: Addison-Wesley, 1978.

Pinker, S., *Language Learnability and Language Development*, Cambridge, MA: Harvard University Press, 1984.