Acquiring languages with movement*

Edward P. Stabler

The position in which a word can occur in a given language is not generally determined by adjacent words, but by structural relations with constituents that can be arbitrarily far away. One prominent idea in the generative tradition is that at least some of these "non-local" dependencies result from the movement of (possibly complex) constituents to various positions where the relevant dependencies are structurally local (but still not necessarily string-local). A consideration of questions, topicalized sentences and other structures in English and other languages makes the hypothesis that constituents get shuffled around initially plausible, but of course the empirical success of the proposal requires careful assessment and comparison with the various close alternatives. In any case, adopting the basic assumptions of generative grammar leads to a particular view of what the most basic properties of lexical items are, and consequently bears on theories about how these properties are determined in language acquisition. In particular, the assumptions of generative grammars do not fit with the structuralist idea that syntactic categories are identified by their distributional properties (Harris, 1951), an idea that has been resuscitated in recent work (Kiss, 1973; Elman, 1993; Mintz, Newport, and Bever, 1995; Cartwright and Brent, 1997). The problem is that two expressions can be similar in their distributions, but differ in the structural configurations in which they occur and in properties that are satisfied non-locally. That is, in a generative framework, a category is identified not by its linear positions in strings of the language, but by its role in the generative structure of the language. A lexical constituent's syntactic properties (like the syntactic properties of any other constituent), are identified not by its neighbors but, one could say, by its deeds.

In this paper, a generative grammar that uses movement as a basic structure-building operation will be formalized in order to define some learning problems. A traditional learning strategy in the "Gold paradigm" is then presented which can identify these generative grammars from examples of structures generated by those grammars. Learning strategies of this kind do not easily extend to more realistic problems, like the problem of identifying the grammar from a sample of possibly ambiguous strings, where the input data is possibly "noisy." In light of these problems, a more flexible learning strategy is proposed based on the same generative framework, a learning strategy that crucially uses the generative potential of the grammar in determining the lexical classification of unknown words. This learner can be regarded from the perspective of "minimal description length" (MDL) theory (Li and Vitányi, 1994; Rissanen and Ristad, 1994) as one that attempts to discover more succinct descriptions of the input, where the notation available for the descriptions is not restricted to linear concatenation of words or categories but is the language of the generative theory.

1 First steps

Gold (1967) established some basic properties of a certain simple model of language learning, where the evidence available to the learner consists of just example strings from a language ("positive text"):

(1) Every finite set of languages can be identified in the limit.

(2) No strict superset of the class of finite languages can be identified in the limit.

*Earlier versions of this material were presented at the University of Potsdam, UCLA, and MIT, where the discussions were very useful.

If lexicons and other "peripheral" properties are separated from the "core grammars" of human languages, then perhaps there are only finitely many core grammars altogether and then their identifiability is given by result (1), as pointed out by Chomsky (1981, p.11) and others. The core language identification problem is still non-trivial when additional constraints are imposed to make the conditions more like those of human language learning. One tradition has explored "parameter-setting" models based on additional assumptions like the following:[1]

(3) a. The parameters of variation among human languages are finite in number.

b. The parameters of variation are binary valued. For example, the language may put the specifier before the head or not; it may put the complement before the head or not; it may require the verb to be in V2 position or not, etc.

c. The parameters are set to some value in the initial state of the learner (the "unmarked" value).

d. Processing a new input sentence sets at most one new parameter.

e. Each parameter has "triggers," where a "trigger" is a sentence that occurs in a language just in case the grammar(s) defining that language have some particular parameter setting.

A different approach to language acquisition is indicated by recent work in linguistic theory. In the languages that are best understood, the possible positions of overt noun phrases and verbs are largely dictated by the verbal inflection, the case marking, and similar things. So then it is a small step to assume that variations in positions of constituents across languages can be attributed to the various requirements of lexical elements, especially the grammatical morphemes. This hypothesis has turned out to be quite fruitful (Emonds, 1985; Borer, 1986; Rizzi, 1986; Rizzi, 1989; Pollock, 1989). Another larger step in the same direction attributes all variation in linguistic structure to lexical variation (Chomsky, 1995). On this perspective, the position of the subject relative to a verb and object in a simple clause is determined by the lexical elements of the clause. Within a single language, the coherence of these properties across lexical items of the same type, to the extent that it exists, might then be attributed to some pressure for a kind of "paradigm uniformity" in the lexicon.

With these assumptions, language acquisition is lexical acquisition. Although this perspective might seem at first to be fundamentally at odds with the parameter-setting models, it is not. The basic observation upon which those models are based can be maintained, namely: while certain fundamental aspects of human languages are universal, others appear to admit of a very limited range of variation. If this is true, and if all variation is lexical, then there is a limited range of variation in certain important lexical properties. A lexical approach to language acquisition could even adopt the assumption that there is a principled distinction between these core properties and the rest. The additional assumptions of particular parameter setting models, on the other hand, assumptions like (3a)-(3e), were never well-supported by the facts. Rather, they are idealizations which make parameter-setting problems easier to study. Interestingly, these are not appropriate idealizations for lexically-based acquisition theories, and so a new perspective on human learning emerges, as will become clear. Natural learning strategies will be proposed here that do not conform to any of (3a)-(3e).

2 Lexicalized grammars with movement

A very simple formal model of minimalist grammar will now be described, loosely based on the proposals of Chomsky (1995, §4) and others. This simplistic model is not a new theory of human language, but just a simple starting point for our investigation of learnability.

The assumption that linguistic variation is lexical dictates that the structure building operations of languages do not vary, so a minimalist grammar is given by its lexicon. The language generated by the

[1] For formal studies of hypotheses like these, see for example Clark (1989), Gibson and Wexler (1994) and the discussions of that work in Berwick and Niyogi (1996) and Frank and Kapur (1996).

lexicon is the set of all the structures that can be built by applying the operations *merge* and *move* to structures in the language.[2]

Lexical items are trees, sometimes simple (consisting of just one node that is both the root and the only leaf of the tree), and sometimes complex. With this assumption, the structure building operations can be defined so that they map trees to trees. The leaves of these trees, the "heads," are complexes of features, some syntactic, some phonetic, some relevant only to interpretation. So for example, the verb `praises` may have a syntactic feature `v` (verb). (Categorial features are in lower case so that capitalization can be used to indicate "strength," as explained below.) Phonetic properties will be indicated by slashes `/praises/`; features relevant only to interpretation will be indicated by parentheses `(praises)`, and the simultaneous presence of non-syntactic features of these types will be represented simply by `praises`. It will be convenient to assume that the features are listed in a sequence, so the simple lexical "tree" for this verb is the following:

```
v praises.
```

A more elaborate structure than a simple sequence will presumably be required in a more sophisticated grammar. To indicate that this verb requires a subject and object, two "selection" features `=d` are added:

```
=d =d v praises.
```

Here, `d` (determiner) is the selected categorial feature.

Since the canonical configuration for head movement is the one that holds between a selecting category and the head of its selected complement, it will be assumed that a verb may not only select the categorial features of the head of its object but may also incorporate the phonetic features of the head of that object `d`. This option is formalized by giving the incorporating verb a "strong" selecting feature, indicated here by capitalizing the category symbol:

```
=D =d v praises.
```

And since incorporated lexical material must be pronounced either before or after the selecting head in time (ignoring the possibility of some sort of "fusion"), let `=D` signify right adjunction of the phonetic material of the selected head, while `D=` signifies left adjunction.[3]

The only other features considered here are those involved in phrasal movement to a specifier position. This movement can be overt – meaning that phonetic features "pied pipe" along with the moved complex of syntactic features – or the movement can be covert, meaning that the complex of phonetic features is left behind. For example, the verb `praise` can assign case to its specifier if it has the feature `+case`, which is added to the lexical item as follows:

```
=d +case =d v praises.
```

If the object of `praise` overtly shifted to get case, the verb would have the corresponding strong case feature:

```
=d +CASE =d v praises.
```

The feature `+case` or `+CASE` triggers the movement of a phrase whose head needs case. The need for case will be indicated by the feature `-case`. So, for example, a proper name could be a lexical item with properties like the following:

[2]Keenan and Stabler (1996) and Keenan and Stabler (1997a) show that this kind of generative definition of a language induces a simple algebraic structure, in terms of which significant universal constraints on the relation between form and meaning can be stated.

[3]The double-bar notation here is inspired by the analogy Koopman (1995) draws between the satisfied requirements of a head and strong chemical bonds.

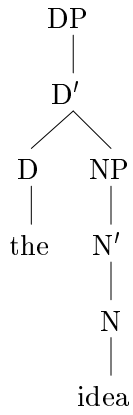
```
d -case john.
```

The features indicate that this is a determiner that needs case, with the phonetic features that will be represented by the notation `/john/` and interpreted features that will be represented by the notation `(john)`.

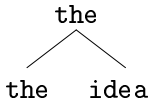
For the purposes of the simple grammars in this paper, the lexical items will have simple trees. Complex lexical trees may be motivated by considerations like those offered in Hale and Keyser (1993), and nothing in the formal model defined here would need to change to accommodate them. The structure building rules, merge and move, apply to trees (complex or not), to yield new trees.

2.1 Merge

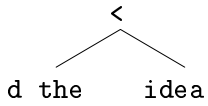
Chomsky (1995) points out that in conventional depictions of determiner phrases like the following, the notation does not make perfectly clear that it is the particular features of head which determine the properties of the phrase:



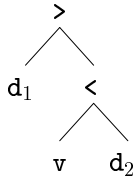
The conventions of X-bar structure also do not allow for the possibility that a single constituent could be both a head and a phrase. Chomsky suggests that a notation like the following might be better:



This step is desirable here, since lexical features are going to determine the whole language, but the notation Chomsky suggests seems to show that the features of `the` occur twice in the structure. To eliminate this confusion, rather than duplicating the head, an "arrow" is placed at every branching node to point down the branch that has the head of the complex. With this convention, the following is a determiner phrase – a bare phrase structure complex whose head has the category `d`:



With this notation, a verb phrase with a complement and specifier (a "VP shell") would be represented with a structure like the following:

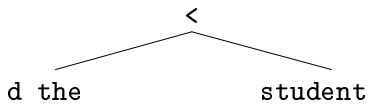


Following the "arrows" down from the root, it is easy to see that this complex has the categorial feature `v`. That is, it is a VP.[4] As usual, $\texttt{d}_1$ is called the "specifier" of the VP – here assumed to always precede the head, and $\texttt{d}_2$ is the "complement" of the VP – always following the head.

The structure-building operations defined here are all feature-driven, even the basic merge operation, in the strong sense that merge can apply only to a pair of trees where the head of the first selects the category of the second, and where the application of merge deletes both features. Furthermore, the features of a lexical item must be checked in order. So, for example, merge applies to the pair of trees:

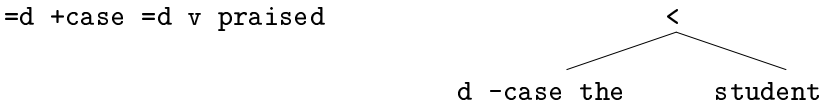
```
=n d the                    n student
```

The result of the operation is the following single tree:

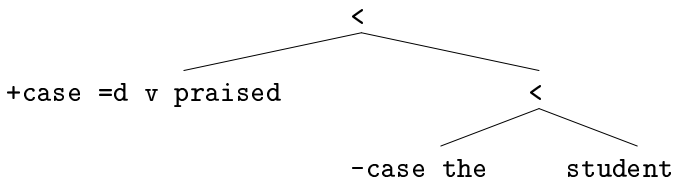


Notice that the operation has deleted exactly 2 features. The selection requirement `=d` has been canceled against the categorial feature `d`.

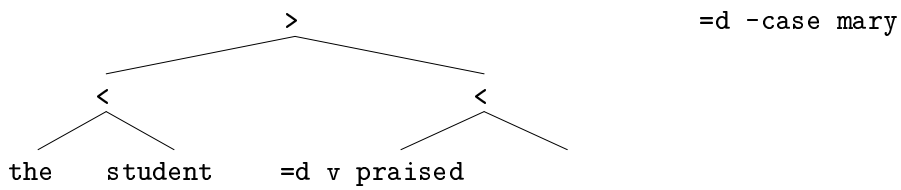
Merge can apply again to the following two trees, the first of which selects a determiner and the second of which has a determiner head:



The result is the following tree. Following the "arrows" from the root shows that this tree has a head with the categorial feature `v`. That is, this result is a VP which has selected its complement.

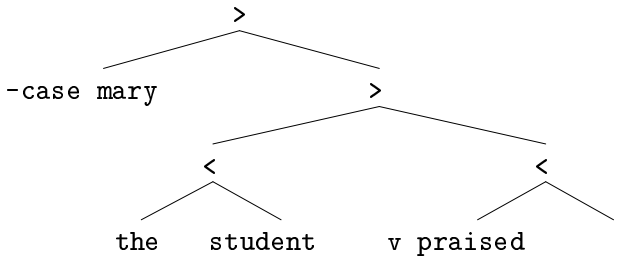


There are two special cases of head movement which require a slightly more elaborate action. First, a complex phrase with an outstanding selection requirement can "select" its outstanding arguments, but these are attached in specifier position rather than complement position. So, for example, merge can apply to the following two trees:



[4] The "arrows" are better understood as indicating a third partial order on the nodes of the tree. In these trees, besides dominance and precedence, every pair of sisters is ordered by a relation that indicate which sister is "projecting over" the other. See Stabler (1997a,b) for further discussion.

The first tree has a head of category `v`, and it has a specifier and an empty complement, but it is still has the `=d` feature which indicates that it wants to select another DP. In this kind of case, where the selector already has a filled complement, merge applies to attach the selected constituent in specifier position, forming a "shell"-like structure, a VP with two specifiers:



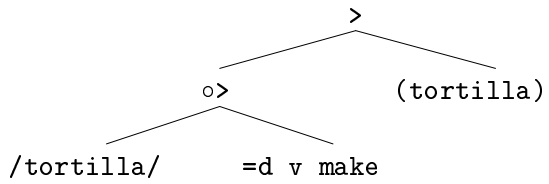
There is one final special case of merge, already mentioned above. In our simple formal system, overt head movement is treated as the "pied piping" of phonetic features along with the selected categorial feature of a complement, triggered by a strong selection feature.[5] Apparent left head adjunction is seen in noun incorporation structures like the following, from Nahuatl, an Uto-Aztecan language of Mexico (Hill and Hill, 1986):

(4) Ni-tlaxcal-chihua
I-tortilla-make

Structures like this are derived by applying merge to trees like the following:

`D= =d v make` `d tortilla`

In the result, the adjunction of phonetic material to heads, these "level 0" elements, is indicated with a special "arrow" `o>`:



Notice that the moved head is interpreted in its original position. The interpreted features `(tortilla)` remain in their original position, but the phonetic features `/tortilla/` have moved. The result is pronounced `/tortilla make/`.

Right head adjunction seems to be indicated by structures like the following, from the language Sora, a Munda language of India (Baker, 1996):

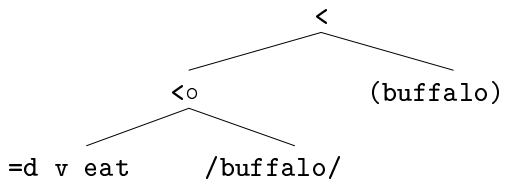
(5) Jom-bɔŋ-t-ɛ-n-ji pɔ
eat-buffalo-NONPAST-3S-INTR-3PS Q

Structures like this are derived by applying merge to trees like the following:

`=D =d v eat` `d buffalo`

[5]This treatment of head movement does not allow movement from the head of the specifier of a complement, or any kind of long distance head movement, and so it may not be quite what we need. However, this approach is very simple, it explains the coincidence of the selection and head movement configurations, and it avoids the problem that head movement, if it is a separate step following merge, violates the simple "extension requirement" on structure building operations, as discussed by Bobaljik and Brown (1997) and others.

The result of the merge operation is a VP in which the phonetic content of the object has right-adjoined to the verb:

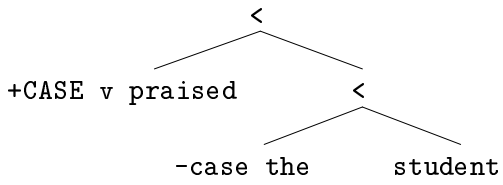


This result is pronounced `/eat buffalo/`.

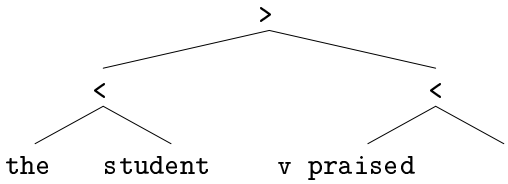
2.2 Move

Having treated head movement as an reflex of merge when it is triggered by a strong selection feature, it only remains to treat phrasal movement, the movement of maximal projections. As one would expect, the maximal projection of a head α is the largest constituent that has α as its head.

Movement applies to a tree like the following, because the head of this tree has a feature `+CASE`, and the tree contains exactly one `-case` element:[6]

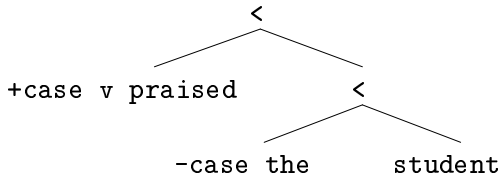


The maximal projection of the unique `-case` head is the right sister of the verb, so the result of applying move to this structure is the following, in which the maximal projection of the `-case` element has moved, canceling the `+CASE` against `-case`:



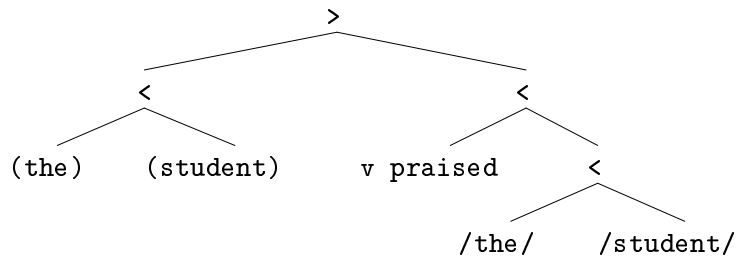
Notice that this treatment of overt movement assumes that all features of the moved constituent are pied piped with the `-case` feature, leaving nothing at all behind except an empty node.

Covert phrasal movement is the same, except that it is triggered by a weak feature `+case` and consequently the phonetic material of the `-case` phrase does not pied pipe along with the rest of the moved structure. Movement applies to the following structure:



[6]The requirement that move applies only to a tree with a `+X` head and with exactly one `-x` subconstituent imposes a simple form of the "shortest link condition." If there were two `-x` features in a phrase, they would compete for the first available `+X` specifier, and only the closest could win. Various "locality" conditions are discussed at some length in Stabler (1997a).

Move leaves the phonetic structure of the `-case` phrase behind, splitting away the syntactic and interpreted features, so we obtain the following structure:



Reading the phonetic material in the order given by the linear precedence in the tree, this phrase is pronounced `/praised the student/`. Reading just the interpreted features, on the other hand, it is interpreted as `((the student) praised)`.

That completes the definition of our minimalist grammars. Notice that each application of a structure building rule deletes exactly 2 features, always the first features in the sequences of features at the heads of the trees involved. Move is sometimes described as involving a copying step and a deletion step, but in this grammar formalism there is no copying operation. Copying duplicates features, increases the "resources" available in the derivation, and the idea here is that this almost never happens in human languages. A single noun phrase can be a complement of one verb, it can receive case once, it is pronounced once. Most grammatical relations have this kind of "bi-uniqueness." The present formalism allows no exceptions to this, no feature duplication or re-use at all. Some duplication or re-use may be required, but it is natural to assume that it will happen in rather special and restricted cases. For the moment, none is allowed.

Move is sometimes also compared to merge. In effect, move merges a tree with a copy of one of its subtrees. The reason that move is not described that way here is to allow the language to be generated as a simple closure. That is, the language is the result of applying the structure building operations to the lexicon and to structures that are built from the lexicon by the structure building rules. The alternative formulation of move as a kind of merge has the problem that the subtree that is moved will often not be a tree that is either in the lexicon or in the set of structures that can be built from the lexicon.

3 A grammar and derivation

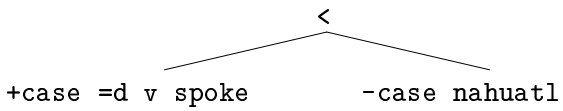
3.1 An English-like language

A first, simple grammar for an infinite English-like language is given by the following 10 lexical items, using `c` for complementizers and `t` for tense:

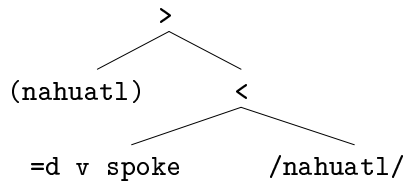
```
=d +case =d v spoke          d -case maria
=d v laughed                 d -case nahuatl
=c +case =d thought          =n d -case some
=t c                         n student
=t -case c                   =v +CASE t
```

With this grammar, move and merge can be applied to obtain infinitely many structures in which all syntactic features have been checked and deleted except for a `c` at the head of the whole complex which indicates that it is a clause. (The `c` could be deleted too when the sentence is integrated into a discourse structure.) The only other features remaining in the structure are the phonetic and semantic features which are presumably interpreted at the respective interfaces. Derivations of such structures will be called "successful." We present one here in full detail.

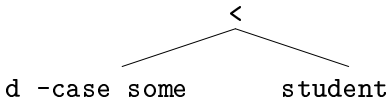
Step 1: Applying merge to lexical items we can obtain:



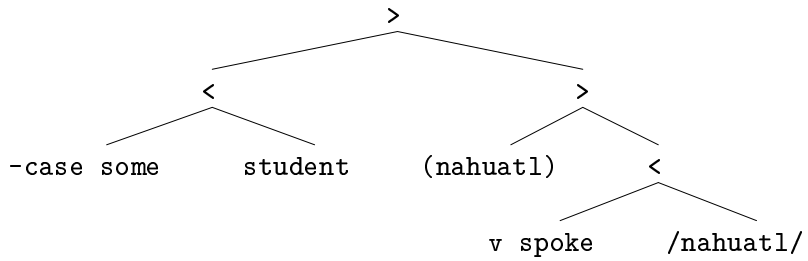
Step 2: Applying move to the result of step 1:



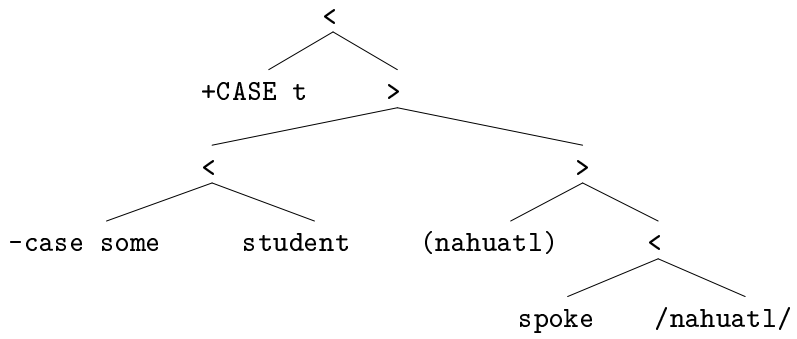
Step 3: Applying merge to lexical items again yields a determiner phrase:



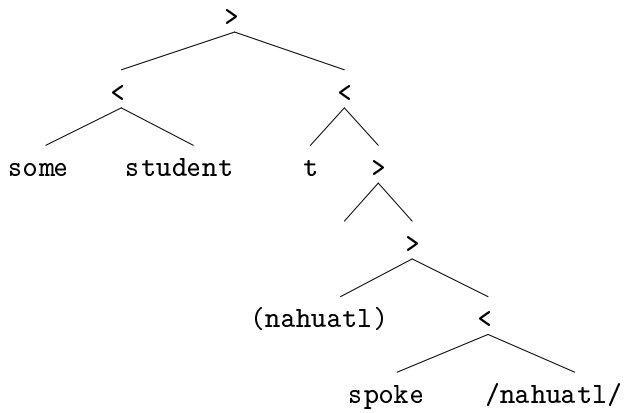
Step 4: Merging the result of step 2 with the result of step 4 yields a VP shell:



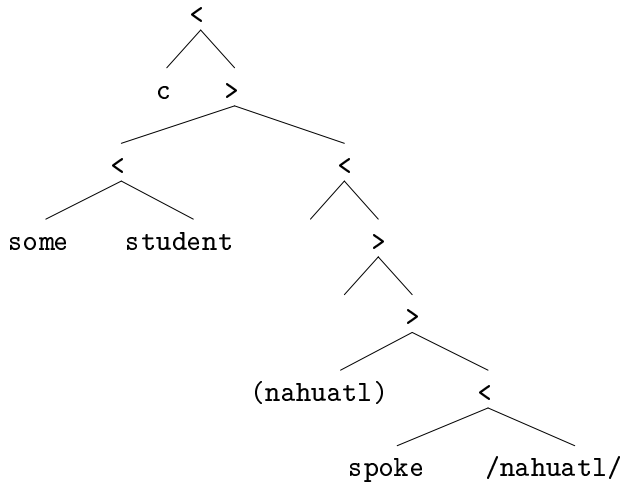
Step 5: Merging a lexical item (tense, which selects v) with the result of step 4:



Step 6: Applying move to the result of step 5:



Step 7: Merging the empty complementizer from the lexicon (the one that does not require case) with the result of step 7:



This completes a successful derivation, a derivation of a structure in which all syntactic features have been deleted except for the root category `c`. Reading off the phonetic features in order yields `/some student spoke nahuatl/`, while the interpreted features appear in the structure `((some student) (nahuatl spoke))`. The lexicon given above allows us to derive infinitely many other structures in which all syntactic requirements have been met, including for example one with the phonetic features `/some student thought maria spoke nahuatl/`.[7]

[7]The derivation of this complex sentence reveals that we are making the simple assumption that sentential complements require case. As Stowell (1981) and many others have argued, this assumption may be incorrect, but this simple grammar will serve for present purposes.

3.2 An SOV language

The pronounced clauses of the previous example have SVO order. SOV order can be obtained by giving transitive verbs strong case features. There are just two verbs that assign case in the previous grammar, so just changing the two case features in those lexical items yields a minimally different grammar for an SOV language:

```
=d +CASE =d v spoke          d -case maria
=d v laughed                 d -case nahuatl
=c +CASE =d thought          =n d -case some
=t c                         n student
=t -case c                   =v +CASE t
```

With this grammar we derive: `/maria nahuatl spoke/` and `/some student maria laughed thought/`.

3.3 A VSO language

Verbs can be attracted to clause-initial position by making the selection features on the tense and complementizers strong. A verb can even be pulled from a lower clause to form a kind of verb cluster if the clausal selection features of the embedding verb are also strong. Changing these four features from the previous example grammar, we obtain:

```
=d +CASE =d v spoke          d -case maria
=d v laughed                 d -case nahuatl
=C +CASE =d thought          =n d -case some
T= c                         n student
T= -case c                   V= +CASE t
```

From this grammar it is easy to derive `/laughed maria/` and `/spoke maria nahuatl/`.[8]

3.4 Elaborations

The minimalist grammars formalized here are quite powerful. It is easy to show that every context free language is definable in this notation, as are many languages that are not context free, and even languages that cannot be defined by any TAG grammar (Cornell, 1996; Stabler, 1997b). Given Gold's result (2), it follows immediately that the class of languages defined by these grammars is not identifiable in the limit from positive text. However, linguistic studies show that these grammars are not powerful enough. It is worth mentioning a few points of this kind, leaving a more careful survey to another place.

In the first place, Chomsky (1995), Ura (1996), Collins (1997) and others have argued that natural languages can be defined more elegantly with more elaborate feature checking regimes. For example, Chomsky does not assume that syntactic features are disjoint from the interpreted and phonetic ones, suggesting that there are two types of features, the $-$interpretable features which are eliminated "at LF", and the +interpretable ones which are not (Chomsky, 1995, pp278-279), where the $-$interpretable syntactic features may have phonological effects.[9] It might also be desirable to allow syntactic operations to check many features in a single step. These schemes are presumably strictly more powerful than the ones proposed in the simple formalism above. In our formal grammars, the features are, in effect, all $-$interpretable, without phonological consequences, and each operation checks and deletes exactly two features.

[8] The other, less common constituent orders can be obtained too, but not by simply varying the strengths of the basic features of this grammar. Other features are needed. One example is given in grammar (17), below, where a topicalized-like OSV order can be derived.

[9] Keenan and Stabler (1997b) show that this classification is probably misnamed. In an adequate semantic theory, it is not likely that literal interpretability is quite the right distinction.

Another elaboration which leads to a strictly more powerful formal framework is one in which there is a wider range of options in determining the "pied piping" of features. In the simple grammars above, a very limited range of variation is determined by the strength of features. A wider range of options is exploited in Stabler (1996) to implement a theory of quantifier raising, but much more radical proposals have appeared in the literature. If Kayne (1994), Mahajan (1995) and others are right that all movement is leftward, then constituents pied-piped to the left are often large enough to make it look like smaller constituents are moving to the right. Some recent and tentative explanations of why things might work this way are now appearing (e.g., Koopman 1996)..

Another respect in which the grammars proposed here could be elaborated is with "transderivational" constraints, constraints that block some derivations on the basis of a comparison with other available derivations. These complicate the picture, and their status remains controversial. They are considered in Stabler (1997a) but will not be discussed here.

4 An infinite class of minimalist grammars, MG_∞

Linguists sometimes assume that there is a fixed, universal set of syntactic categories, and even that clause structure is essentially identical across languages. If this kind of restriction were really plausible on empirical grounds, it could be captured in the present framework with restrictions on the range of possible categories. A very simple model of this kind is sketched here.

Let MG_∞ be the class of minimalist grammars with the properties (6a)-(6d):

(6) a. In all languages the root category of the good sentences is `c`. (All other syntactic features are checked and deleted.)

b. The possible sequences of syntactic features in each lexical item are just the following 24:

`=t c`	`=T c`	`T= c`		
`=t c -case`	`=T c -case`	`T= c -case`		
`=v +case t`	`=v +CASE t`			
`=V +case t`	`=V +CASE t`	`V= +case t`	`V= +CASE t`	
`=d +case =d v`	`=c +case =d v`	`=C +case =d v`	`C= +case =d v`	`=d v`
`=d +CASE =d v`	`=c +CASE =d v`	`=C +CASE =d v`	`C= +CASE =d v`	
`d -case`	`=n d -case`	`n`		

This set has basically just the categories already seen in the first simple grammar of §3.1, except that the features `+case, =c, =v`, and `=t` are allowed to be strong or weak.

c. No bound is imposed on the pronounced and interpreted features of the lexical items. They can be regarded as coming from some infinite set V_∞.

d. Every lexical `d`, `n`, and `v` is phonetically nonempty, but other categories may be phonetically empty.

A minimalist grammar is just a set of lexical items, so any set of lexical items whose elements meet these conditions is in MG_∞. Let $\mathcal{ML}_\infty$ be the collection of languages (the sets of pronounced strings of good sentences) defined by elements of MG_∞.

With these definitions, it is easy to establish the following facts:

(7) a. $\mathcal{ML}_\infty$ does not contain all finite languages over the vocabulary V_∞.

b. $\mathcal{MG}_\infty$ is infinite (because V_∞ is), and so is $\mathcal{ML}_\infty$.

c. $\mathcal{ML}_\infty$ contains some finite and some infinite languages.

d. $\mathcal{ML}_\infty$ contains SOV, SVO and VSO languages.

e. If $G_0, G_1 \in \mathcal{MG}_\infty$, then so are $(G_0 \cup G_1)$ and $(G_0 \cap G_1)$.

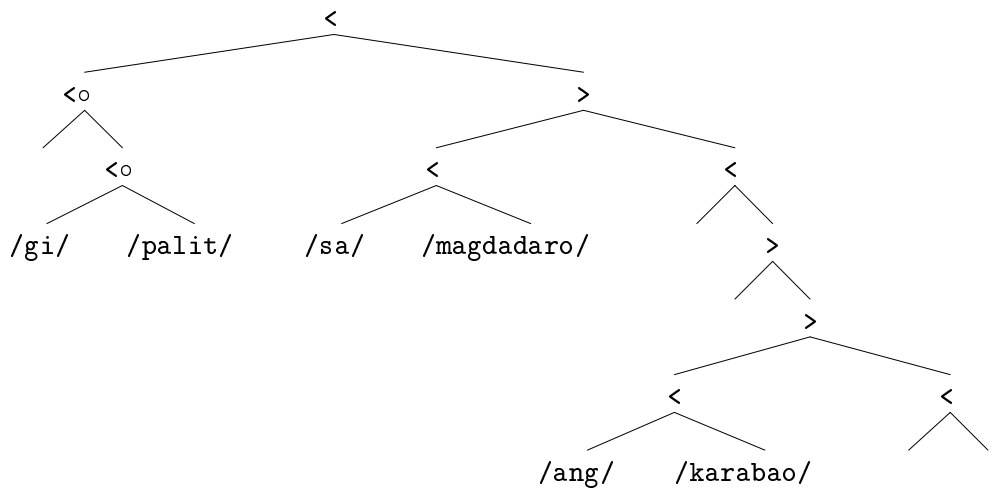
f. If $G_0 \subseteq G_1$ then $L(G_0) \subseteq L(G_1)$.

g. Putting two grammars together ($G_0 \cup G_1$) will often yield structures that are not generated by either grammar by itself.[10]

With property (7a), the simple negative result (2) from Gold does not apply. And with property (7b), the trivial positive result (1) does not apply either.

5 Learning MG_∞ from structures

One strategy for making a learning problem easier is to enrich the evidence. Although (2) entails that context free languages cannot be learned from strings, a certain subset of these languages can be learned from phrase structure "skeletons" – derivation trees with all categorial information removed (Sakakibara, 1992). And a certain class of categorial grammars can be identified in the limit from derivation trees which contain no no categorial information but indicate which elements are arguments and which functions (Kanazawa, 1994; Kanazawa, 1996). In a similar spirit, let's consider first the problem of identifying grammars in the restricted class MG_∞ from "skeletons" – derived structures from which all syntactic and interpreted features are deleted. So, for example, imagine that the learner gets as input a text of skeletons like the following, a skeleton which is pronounced `/gi palit sa magdadoro ang karabao/`.



This skeleton may look forbidding at first, but it is easy to see that it provides full information about the syntactic features of its lexical elements. We quickly sketch how this information can be determined.

By (6a), the learner can assume that the root category of this tree is `c`. It is easy to see that the root category of the skeleton has no phonetic features, and that it has right adjoined a complex `/gi palit/` whose head must be tense `t`, given the categorial restrictions in (6b). So the lexical complementizer here can only be:

```
=T c.
```

Clearly, the head of the adjoined tense complex is shown with the phonetic features `/gi/`, and it has a right adjoined constituent which, given the categorial restrictions in (6b), can only be the verb. Every lexical element of category `t` selects `v`, but this one strongly selects and right adjoins it, so it has the feature `=V`. Since the tense phrase must be the complement of the root `c`, its original position in the tree is easily identified, where its projection includes an overtly filled specifier, which can only happen if the

[10] These new "code-switching" structures are immediately generated without any special additional mechanisms. Since lexical features can impose rather complex requirements on their environment, we can also avoid the false prediction that, for example, a code-switcher will freely replace a verb in one language by a verb in another. A minimalist account of code-switching is developed along these lines by MacSwan (1997).

tense element has the feature +CASE. The structure by itself does not reveal what the semantic value of this element is, but letting (gi) stand for that unspecified value as usual, the lexical tense head can only be:

=V +CASE t gi.

Turning now to the verb, its pronounced features have already been found: it is /palit/, and of course it is the complement of the tense phrase. There, it can be seen to have two arguments, a subject and a complement of some kind (though the complement has shifted to specifier position). The categorial restrictions require that the subject is a d, and that subject is the only thing that can be in the specifier position of the t phrase, /sa magdadoro/. The complement /ang karabao/ is overt in the specifier of the verb, which means that the verb must have the feature +CASE. Verb complements may be either a c or a d, but here it must be the latter, since no clause can have a structure as simple as the one /ang karabao/ has. So, the lexical verb in this derivation can only be:

=d +CASE =d palit.

Continuing with the same reasoning, we can see that the lexical items used to construct the arguments must be:

=n d sa
=n d ang
n magdaduro
n karabao

With these determinations, all the lexical elements in the structure have been identified. The skeleton is revealed as a clause with VSO constituent order.[11] The reasoning used here clearly extends to structures of arbitrary complexity. Some details need to be filled in to make the strategy perfectly clear, but it should be plausible already that, processing each structure in the text with this kind of reasoning, the learner will gradually accumulate all the lexical entries of the language, identifying the complete grammar in the limit (assuming that the lexicon is finite and that after seeing finitely many skeletons from the text, the learner will have seen all the lexical items, and keeping in mind that the semantic features will be left unspecified).

It is worth pausing here to reflect on the basic properties of this learning strategy. The categorial assumptions (6b) in the definition of MG_∞ are obviously much too simple for any realistic theory of human languages. We could begin to elaborate them while keeping track of whether the elaborated class remains identifiable from texts of skeletons. We could also investigate the conditions under which the grammar could be identified with less than full skeletons: perhaps from strings with prosodically indicated bracketings, or from strings with semantic information. However, other properties of the learner suggest that much more fundamental adjustments are needed if we want to approximate the human learner in a fruitful way.

A first problem. The class MG_∞ cannot be identified from positive texts of strings. This immediately follows from the fact that there are different grammars which generate exactly the same strings. For example, the SVO grammar given in §3.3 generates the same strings as the grammar which is identical except that the case features on the verb are weak, +case.

[11] This example structure was inspired by the following sentence from the Philippine language Cebuano (Bell, 1983):

Gi-palit sa magdaduro ang karabao
OBJ-buy GEN farmer NOM buffalo
'The buffalo was bought by the farmer'

The categorial system of MG_∞ is obviously not rich enough to give a proper treatment of case markers or to distinguish simple tense markers from the rich aspect and voice system found in languages like this (here we see the objective voice marker gi-).

One might assume that this problem would be removed if that grammar were enriched with adverbs or negation or any other constructions which could reveal whether the object had shifted, but in fact this would not remove the problem. Suppose that the grammar of §3.3 were enriched so that in addition to `/spoke maria nahuatl/` it generates some string s with the verb `spoke` in a way that reveals that the object has not shifted. The problem is that this structure s would not entitle the learner to conclude that `spoke` does not shift its argument, but only that the verb `spoke` does not shift its argument in s. There could well be another verb that has the same phonetic features which does shift its argument, and it could be that this other verb is the one we heard in `/spoke maria nahuatl/`. That is, the existence of homonymy in human languages makes it risky to reason across occurrences, and so in the Gold paradigm, identifiability from texts of strings will be lost.

The basic point is: the simple learner described above never needs to reason across various occurrences of a lexical item in order to determine its properties, but if the data is just strings, there is a serious problem here. Some lexical items have properties that are not revealed by any single occurrence in a string. For anyone who has ever tried to describe a human language, the idea of trying to determine the properties of any verb from a single occurrence of that verb in a string is ludicrous. It is no wonder that so many learning models either assume that there is no homonymy or that the learner is given explicit information (e.g. identifying semantic properties, or elaborate syntactic structures) to indicate which occurrences can be related.

A second problem. Given noisy input, a learner like the one sketched above can make a mistake about the grammar (that is, it will posit an incorrect lexical entry), and of course every mistake is permanent since no lexical items are ever retracted. An error about a basic grammatical morpheme would be most serious. For example, if in a collection of some thousands of SVO inputs, there were mistaken VSO input, the learner would add lexical items which would allow the verb to be fronted in any sentence!

This concern is clearly related to the previous one. In both cases, the problem is that the learner not reasoning in an appropriate way about collections of related structures. The oddness of a spurious VSO input in an SVO corpus would not be noticed.

A third problem. Very few lexical items are perfectly equivalent in terms of their roles in the grammar. That is, at the finest level of detail, very many categories are distinguished (Levin, 1993). There is no good reason to assume that there is a linguistically principled bound on the number of syntactic distinctions that can be drawn, or that the language learner exploits that kind of bound in the attempt to determine the properties of unknown lexical items.

A fourth problem. Language learners do not immediately analyze everything down to the atomic generators, the smallest lexical units which could yield the input. Rather, boundaries and generalizations are often unnoticed by human learners, while idioms and subtle connotations of particular phrases are easily noticed. An impressionistic observation like this one would not be an immediate cause of concern except that it is apparently related to the previous points, and that anything of this sort is rather surprising in models with categories given as part of the innate universal base.

6 Beyond MG_∞

Suppose that the categorial restrictions on MG_∞ are withdrawn, and the learner must attempt to identify the categories of minimalist grammar from the evidence presented. That is, the learner's grammars can be defined as before with categories that can be selected strongly or weakly, but the categories are not specified in advance. Furthermore, there can be weak and strong triggers for phrasal movement, but the these are not specified either. With this system, as mentioned in §3.4 above, many languages can be defined: all the context free languages and many non-context free languages. So of course this class of languages is not identifiable in the limit from positive text. Every finite set is definable, so the learner can never be secure in making any generalization beyond what has been seen. In this context, though, where our grammar notation can provide many different grammars compatible with the input that has been seen

at any point, we can get the learner to generalize by providing some simplicity metric on the available hypotheses, where the simpler hypotheses are preferred, as in Chomsky (1965, §1.7), for example. This idea is deployed in Berwick (1982) and much recent work. The simpler hypotheses will be the ones that capture generalizations in the data. This does not quite solve the problems listed just above, though, unless the measure is sensitive to how much of the data each part of the hypothesis covers. To notice when the uses of a given pronounced form fall into two different kinds of constructions, indicating a possible homonymy, and hence a license to reason across the related uses but not across all uses, a sensitivity to quantity of evidence is required. To avoid getting misled by relatively infrequent "noise," to leave expressions relatively unanalyzed until they seem common enough to prompt the recognition of a regularity, some sensitivity to quantity is again required. The relevant simplicity metric should not be defined over grammars, but over the grammars together with the evidence they are intended to cover. This is the basic idea of "minimal description length" (MDL) approach.

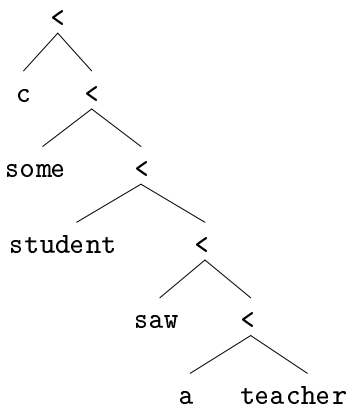
In the MDL approach, the learner adopts a generalization when it provides a significant simplification of the representation of the grammar and the data. Suppose, for example, that the learner hears `/some student saw a teacher/`. There is not yet any reason to assume that these words fall into general categories at all. Each word w can be treated as the unique member of its own category $\mathtt{x}_w$. In the absence of prosodic cues, the first input could be analyzed with the trivial grammar, letting `c` here be just the name of the "initial category:"

(8)

$=\mathtt{x}_{some}$ c	$=\mathtt{x}_{student}$ $\mathtt{x}_{some}$ some
$=\mathtt{x}_{saw}$ $\mathtt{x}_{student}$ student	$=\mathtt{x}_{a}$ $\mathtt{x}_{saw}$ saw
$=\mathtt{x}_{teacher}$ $\mathtt{x}_{a}$ a	$\mathtt{x}_{teacher}$ teacher

This grammar does not "generalize" at all from a single input. It derives just one structure:

(9)



Since this is the only successful derivation, nothing else is required to describe the input.

The challenge is to get from this starting point to a grammar that is more like the one in §3.1 above, on the basis of sample strings from the language. In particular,

(10)
a. What prompts the recognition that `student` and `teacher` are in the same category, or even that two different occurrences of `student` are in the same category?
b. What prompts the recognition that `some student` is a constituent?
c. What prompts the recognition that constituents move (to get case, to form questions, ...)?

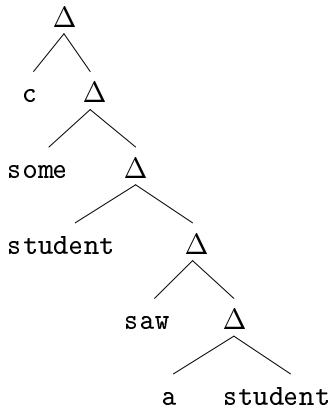
These questions are considered in turn.

Suppose that after hearing `/some student saw a teacher/`, the learner hears `/some student saw a student/`. One possible hypothesis about this input is that it is derived from the following grammar:

(11)

=x$_{some}$ c	=x$_{some2}$ c
=x$_{student}$ x$_{some}$ some	=x$_{student2}$ x$_{some2}$ some
=x$_{saw}$ x$_{student}$ student	=x$_{saw2}$ x$_{student2}$ student
=x$_{a}$ x$_{saw}$ saw	=x$_{a2}$ x$_{saw2}$ saw
=x$_{teacher}$ x$_{a}$ a	=x$_{student3}$ x$_{a2}$ a
x$_{teacher}$ teacher	x$_{student3}$ student

This grammar generates two trees. We can specify the tree just heard by specifying the derivation. Abbreviating each derived tree by Δ and each lexical tree by its English orthography, this derivation has the following structure, a structure that happens in this case to be isomorphic to the result of the derivation which labels the root of the derivation tree:



This derivation is determined by one binary lexical choice.[12]

There are many other grammars which could be used to describe the two sentences above. Taking one small step, consider the following grammar:

(12)

=x$_{some}$ c	=x$_{student}$ x$_{some}$ some	
=x$_{saw}$ x$_{student}$ student	=x$_{a}$ x$_{saw}$ saw	
=x$_{2}$ x$_{a}$ a	x$_{2}$ teacher	x$_{2}$ student

By any reasonable measure, this grammar is simpler than (11): it has fewer categories; it has fewer lexical items; it has fewer features in the lexicon altogether. This grammar still generates only two derivation trees, determined by a single lexical choice. So a learner that aims to minimize both hypothesis complexity and data description will clearly prefer to assume non-homonymy in the inputs unless there is something to indicate otherwise, and to merge categories of elements that distinguish minimal pairs of derivations.

Consider the following bolder hypothesis about the language:

(13)

=x$_{saw}$ c	=x$_{2}$ =x$_{2}$ x$_{saw}$ saw
=x$_{3}$ x$_{2}$ some	=x$_{3}$ x$_{2}$ a
x$_{3}$ student	x$_{3}$ teacher

This grammar is even simpler than the previous one: fewer categories, fewer lexical items, fewer features in the whole lexicon altogether. This grammar allows all three occurrences of `student` in the two input sentences to be treated as having the same category. This is possible when the simple head-complement chain of the earlier hypotheses is abandoned. However, it is important to notice that capturing this generalization comes with a price in data description. Because this grammar defines not 2 successful

[12] An appropriate method of specifying particular derivations and their choice points can be provided based on derivation-traversal methods of the sort described in Stabler (1997c) and Stabler (1997a).

derivations, but 4, specifying the particular input string just heard, `/some student saw a student/`, now requires twice as many bits of information.

The MDL learning strategy selects a hypothesis that minimizes both hypothesis complexity and data description. This last grammar shows how these trade off against each other. A tendency to generalize leads to more complex data descriptions, but resisting generalization leads to large, ad hoc hypotheses like (11). The success of the learning strategy depends on keeping the delicate balance that minimizes the hypothesis and data description taken together, and so it depends on the the the complexity measures of the hypothesis and data description. For the moment it suffices to observe that an MDL learner (using any natural measure of grammar and description complexity) will tend to presume non-homonymy in the inputs unless there is something to indicate otherwise, and to merge the categories of elements that distinguish minimal pairs of derivations. And on the other hand, minimizing the cost of describing the input data (in terms of its derivation) keeps the learner from overgeneralizing, putting everything in the same category to obtain the simplest possible grammar. In these first simple examples one can see the outlines of a possible response to questions (10a) and (10b).

The third question (10c) is a more challenging. Will MDL principles ever lead the learner to assume that some lexical elements have features that trigger movement? In fact, this can happen quite easily. To take a simplistic example, consider the options for generating strings like `/mary will meet sue/` and `/will mary meet sue/`. The right word orders can be generated by grammars like (14) or (15):

(14)

=x$_{mary}$ c	=x$_{will2}$ c
=x$_{will}$ x$_{mary}$ mary	=x$_{mary2}$ x$_{will2}$ will
=x$_{meet}$ x$_{will}$ will	=x$_{meet}$ x$_{mary2}$ mary
=x$_{sue}$ x$_{meet}$ meet	
x$_{sue}$ sue	

(15)

=x$_{will}$ c	=x$_{will2}$ c
=x$_2$ =x$_{meet}$ x$_{will}$ will	=x$_{meet2}$ x$_{will2}$ will
=x$_2$ x$_{meet}$ meet	=x$_2$ x$_2$ meet2
x$_2$ mary	x$_2$ sue

The categories in these grammars are not likely to fare very well when the learner gets more input, but more importantly, these grammars never get proposed because there are more succinct grammars for the same sentences, ones in which `c` strongly selects `will`:

(16)

=x$_{will}$ c	=X$_{will}$ c
=x$_{meet}$ =x$_2$ x$_{will}$ will	
=x$_2$ x$_{meet}$ meet	
x$_2$ mary	x$_2$ sue

Grammar (16) has fewer categories, fewer lexical items, and fewer features altogether than either of (14) or (15). Head movements will be proposed when they allow this kind of succinctness without inducing excessive overgeneralization.

Phrasal movements can similarly lead to economy of expression without unacceptable overgeneration. Notice, for example, that a grammar like the following which minimally extends (13) with `+F` and `-f` elements will get `a teacher, some student saw`, and `a student, some student saw`, and other forms with fronted determiner phrases:

(17)

=x$_{saw}$ c	=x$_{saw}$ +F c	=x$_2$ =x$_2$ x$_{saw}$ saw
=x$_3$ x$_2$ some -f	=x$_3$ x$_2$ a -f	
x$_3$ student	x$_3$ teacher	

The alternative grammars that generate these same sentences without phrasal movement are more complex.

These examples are meant only to suggest that simple and natural assumptions about representation complexity seem to come surprisingly close to providing the right balance between generalization and

conservatism. The measures based simply on counting categories, lexical items and features suffice to make a initial case for the promise of this kind of learning strategy. Of course, the proof will be in the pudding, but the preliminary case for these methods looks promising. Ongoing studies on learning artificial languages from generated samples suggest that better results can be obtained by using natural information-theoretic measures of complexity, adapting the ideas of (Cartwright and Brent, 1997) and others to this quite different minimalist setting, where the lexical classification determines the structure of the whole language. With such information-theoretic measures, the frequency with which expressions occur in the data becomes relevant, driving the learner to work hardest on finding generalizations about frequently occurring data.

It could be that eliminating all constraints on the categorial system provides more flexibility than we really need. For example, it could be that the "functional categories" and other properties of the system are innately given. It is also plausible that human learners use more than just distributional reasoning: they use prosodic and possibly also semantic cues even at the earliest stages. These things could clearly be accommodated into an MDL framework like the one sketched here. There is a need, though, to understand what can be revealed by simple distributional reasoning in generative frameworks.

7 Conclusions

Certain ideas from recent syntax are easily formalized in the "minimalist grammars" that have been introduced recently (Stabler 1996, 1997a,b,c; Cornell 1996). These grammars use universal structure building rules, with all linguistic variation attributed to lexical properties, and they are simple enough to allow easy study of learning problems. Language learning in this framework is just identifying the syntactic properties of lexical items. A traditional approach can be taken, assuming that the learner gets very rich data, not just strings but rich structural representations from which to infer the grammar of the community. However, serious problems arise when more realistic assumptions about the data are made. In particular, the learner then needs to be able to reason across various occurrences of a word, something that requires care in the presence of homonymy. This problem is related to various others, such as the need to model a certain degree of insensitivity to noise in the data. In light of these problems, a slightly more flexible learner is proposed here. This learner has the potential to represent a class of languages that is not identifiable in the limit, but a pressure to generalize appropriately is imposed using a "minimal description length" criterion. Generalization is based on an assessment of the simplification achieved. The measure of simplification includes both grammar complexity and the complexity of representing the data, where the data is not simply a set but a sequence in which input strings may occur with various frequencies. In the generative framework defined here, the lexical classification of an element does not depend just on the adjacent elements in the string, nor does it depend on everything in the string. Rather, it depends on structurally defined features that may be arbitrarily far away.

References

Baker, Mark. 1996. *The Polysynthesis Parameter.* Oxford University Press, NY.

Bell, Sarah J. 1983. Advancements and acensions in Cebuano. In David M. Perlmutter, editor, *Studies in Relational Grammar.* University of Chicago Press, Chicago.

Berwick, Robert C. 1982. *Locality Principles and the Acquisition of Syntactic Knowledge.* Ph.D. thesis, Cambridge, Massachusetts, Massachusetts Institute of Technology.

Berwick, Robert C. and Partha Niyogi. 1996. Learning from triggers. *Linguistic Inquiry*, 27:605–622.

Bobaljik, Jonathan David and Samuel Brown. 1997. Interarboreal operations: head movement and the extension requirement. *Linguistic Inquiry*, 28:345–356.

Borer, Hagit. 1986. I-subjects. *Linguistic Inquiry*, 17:375–416.

Cartwright, Timothy A. and Michael R. Brent. 1997. Syntactic categorization in early language acquisition: Formalizing the role of distributional analysis. *Cognition*, 62:121–170.

Chomsky, Noam. 1965. *Aspects of the Theory of Syntax.* MIT Press, Cambridge, Massachusetts.

Chomsky, Noam. 1981. *Lectures on Government and Binding.* Foris, Dordrecht.

Chomsky, Noam. 1995. *The Minimalist Program.* MIT Press, Cambridge, Massachusetts.

Clark, Robin. 1989. On the relationship between the input data and parameter setting. In *Proceedings of the North East Linguistic Society, NELS 19*, pages 48–62.

Collins, Chris. 1997. *Local Economy.* MIT Press, Cambridge, Massachusetts.

Cornell, Thomas L. 1996. A minimalist grammar for the copy language. Technical report, SFB 340 Technical Report #79, University of Tübingen. Available at http://www.sfs.nphil.uni-tuebingen.de/~cornell/.

Elman, Jeffrey L. 1993. Learning and development in neural networks: The importance of starting small. *Cognition*, 48:71–99.

Emonds, Joseph E. 1985. *A Unified Theory of Syntactic Categories.* Foris, Dordrecht.

Frank, Robert and Shyam Kapur. 1996. On the use of triggers in parameter setting. *Linguistic Inquiry*, 27:623–660.

Gibson, Edward and Kenneth Wexler. 1994. Triggers. *Linguistic Inquiry*, 25:407–454.

Gold, E. Mark. 1967. Language identification in the limit. *Information and Control*, 10:447–474.

Hale, Kenneth and Samuel Jay Keyser. 1993. On argument structure and the lexical expression of syntactic relations. In Kenneth Hale and Samuel Jay Keyser, editors, *The View from Building 20: Essays in Linguistics in Honor of Sylvain Bromberger.* MIT Press, Cambridge, Massachusetts, pages 53–109.

Harris, Zellig S. 1951. *Methods in Structural Linguistics.* Chicago University Press, Chicago.

Hill, Jane H. and Kenneth C. Hill. 1986. *Speaking Mexicano.* The University of Arizona Press, Tucson, Arizona.

Kanazawa, Makoto. 1994. *Learnable Classes of Categorial Grammars.* Ph.D. thesis, Stanford University.

Kanazawa, Makoto. 1996. Identification in the limit of categorial grammars. *Journal of Logic, Language, and Information*, 5:115–155.

Kayne, Richard. 1994. *The Antisymmetry of Syntax.* MIT Press, Cambridge, Massachusetts.

Keenan, Edward L. and Edward P. Stabler. 1996. Abstract syntax. In Anne-Marie DiSciullo, editor, *Configurations: Essays on Structure and Interpretation*, pages 329–344, Somerville, Massachusetts. Cascadilla Press. Conference version available at http://128.97.8.34/.

Keenan, Edward L. and Edward P. Stabler. 1997a. *Bare Grammar.* CSLI Publications, Stanford University. Cambridge University Press, NY. forthcoming.

Keenan, Edward L. and Edward P. Stabler. 1997b. Syntactic invariants. In *6th Annual Conference on Language, Logic and Computation*, Stanford.

Kiss, George R. 1973. Grammatical word classes: A learning process and its simulation. *Psychology of Learning and Motivation*, 7:1–41.

Koopman, Hilda. 1995. On verbs that fail to undergo V-second. *Linguistic Inquiry*, 26:137–163.

Koopman, Hilda. 1996. The spec-head configuration. Syntax at Sunset: UCLA Working Papers in Syntax and Semantics, edited by Edward Garrett and Felicia Lee.

Levin, Beth. 1993. *English Verb Classes and Alternations*. Chicago University Press, Chicago.

Li, Ming and Paul Vitányi. 1994. Inductive reasoning. In Eric Ristad, editor, *Language Computations*. American Mathematical Society, Philadelphia.

MacSwan, Jeff. 1997. *A Minimalist Approach to Intrasentential Code Switching: Spanish-Nahuatl Bilingualism in Central Mexico*. Ph.D. thesis, UCLA. Available at http://phonetics.ling.ucla.edu/.

Mahajan, Anoop. 1995. Universal grammar and the typology of ergative languages. In A. Alexiadou and T.A. Hall, editors, *Syntactic Typology and Universal Grammar*. John Benjamins.

Mintz, T.H., E.L. Newport, and T.G. Bever. 1995. Distributional regularities of grammatical categories in speech to infants. In *Proceedings of the North East Linguistic Society, NELS 25*.

Pollock, Jean-Yves. 1989. Verb movement, universal grammar, and the structure of IP. *Linguistic Inquiry*, 20:365–424.

Rissanen, Jorma and Eric Ristad. 1994. Language acquistion in the MDL framework. In Eric Ristad, editor, *Language Computations*. American Mathematical Society, Philadelphia.

Rizzi, Luigi. 1986. Null subjects in Italian and the theory of pro. *Linguistic Inquiry*, 17:501–557.

Rizzi, Luigi. 1989. On the format for parameters. *Behavioral and Brain Sciences*, 12:355–356.

Sakakibara, Yasubumi. 1992. Efficient learning of context-free grammars from positive structural examples. *Information and Computation*, 97:23–60.

Stabler, Edward P. 1996. Computing quantifier scope. In Anna Szabolcsi, editor, *Ways of Scope Taking*. Kluwer, Boston.

Stabler, Edward P. 1997a. *Computational Minimalism: Acquiring and parsing languages with movement*. Basil Blackwell, Oxford. Forthcoming.

Stabler, Edward P. 1997b. Derivational minimalism. In Christian Retoré, editor, *Logical Aspects of Computational Linguistics*. Springer-Verlag, NY. Forthcoming. Draft available at http://128.97.8.34/.

Stabler, Edward P. 1997c. Parsing and generation for grammars with movement. In Robert C. Berwick, editor, *Principle-based Parsing: From theory to practice*. Kluwer, Boston. Forthcoming. Draft available at http://128.97.8.34/.

Stowell, Tim. 1981. *Origins of Phrase Structure*. Ph.D. thesis, Massachusetts Institute of Technology.

Ura, Hiroyuki. 1996. *Multiple Feature-Checking: A Theory of Grammatical Function Splitting*. Ph.D. thesis, Massachusetts Institute of Technology.