Learning Feature-Value Grammars from Plain Text

Tony C. Smith
Department of Computer Science, University of Waikato
Hamilton, New Zealand
tcs@cs.waikato.ac.nz

Abstract

This paper outlines preliminary work on learning feature-value grammars from from plain text. Common suffixes are gleaned from a word suffix tree and used to form a first approximation of how regular inflection is marked. Words are generalised into lexical categories according to regularities in how these suffixes appear in trigram contexts. The categories are expressed as a lexical feature whose value is given by the most frequent suffix for similar trigrams. The trigrams are subsequently used to infer agreement dependencies which are captured through the creation of additional feature structures. Agreement and linear precedence are preserved through the iterative creation of unification rules for pairs of terms.

1 Motivation

Unification grammars (UGs) have become the established formalism for natural language understanding systems, primarily because of their clean denotational semantics and their ability to capture complex grammatical constraints through feature dependencies (Uszkoreit & Zaenen, 1996). But engineering even modestly sized UGs can take a very long time, making the idea of constructing a comprehensive, robust, competent UG by hand virtually intractable. Recent advances in stochastic language modeling, however, have made it possible to incorporate statistical information into UGs (Abney, 1996; Smith & Cleary, 1997), thus giving access to the complexity estimates now widely regarded as essential for automatically learning adequate grammars from positive data alone. But this still leaves open the question of exactly how such learning can be achieved for UGs.

A probabilistic unification grammar (PUG) has three principal components: 1) a context-free account of linear precedence relations, 2) a set of features for expressing grammatical dependencies, and 3) probability distributions for the rules and features. Methods for unsupervised learning of the first and last of these components have already been suitably worked out. For example, the context-free description can be addressed with solutions borrowed from work in learning PCFGs (Jelinek et al, 1992; Chen, 1996), and the distribution can be estimated by training on sample data (Eisele, 1994; Brew, 1995). The outstanding problem then is how to derive a satisfactory set of features in the absence of overt semantic information.

This paper describes preliminary work aimed at learning a Feature-Value Grammar from plain text. It is based on the generally held notion that syntactic agreement and morphological inflection are closely related (Abney, 1987; Fukui & Speas, 1986). Morphological clues about inflection are gleaned from the vocabulary of a language using a word suffix tree. Common suffixes are assumed to identify related syntactic elements undergoing the same inflectional process. Broad generalisations over enclosing trigram contexts are used to categorise words in terms of common suffixes, and prior contexts are subsequently used to identify agreement dependencies, which are captured through the creation and projection of feature structures. Linear precedence relations and the agreement constraints are thereafter expressed using a unification formalism.

2 Finding features

A UG encodes lexical properties as feature structures (specifying such things as part-of-speech, number, tense, person, thematic role, etc.) whose values percolate up through a subsumption hierarchy by the process of unification (Sanfilippo, 1993). Syntactic constraints are imposed by forcing agreement between features of grammatically related structures.

Kazman (1994) argues that features correspond to semantic properties associated with thematic cat-

egories (e.g. nouns, verbs and adjectives) and that learning syntax is equivalent to figuring out how these properties impose constraints on the functional categories (e.g. determiners, auxiliaries, and complementizers) of a particular language. This study takes the slightly stronger position that the process by which thematic and functional categories are combined is mediated by morphological inflection. Like Kazman's system, *Babel*, the focus is on the role of inflectional affixes in the acquisition of agreement. But unlike *Babel*, which makes inferences over semantically related words identified through set operations on input already tagged with attributes, this work addresses feature identification as a bootstrapping problem—where feature structures are learned at the same time as the constraints that they impose. Thus the input is plain text.

Word suffix tree

The first objective is to detect when and how inflection is manifest. This is addressed through generalisation on a word suffix tree (WST) constructed for the vocabulary of the language. A WST is a derivative of a letter-based multiway trie built from an ordered set of words. Each distinct sequence of characters along a path in the trie is collapsed into a single node, resulting in a WST for which all leaf nodes are common suffixes to the prefix terminated by their parent node (Andersson, 1996). A sample portion of a WST is shown in Figure 1. Note that the symbol *$* is a kind of NULL suffix, which shows that the parent node is itself a suffix and thus corresponds to the end of an actual word. It follows that its leaf nodes correspond to genuine morphological suffixes.

Inflectional suffixes

Given that regular inflection is largely realised through suffixion on root categories, a first approximation of these categories may be given by assigning a common lexical identity to words that share the same set of suffixes. That is, it is assumed that words which inflect in the same ways likely belong to the same syntactic category. Clearly not all suffixes are inflectional. Therefore, some general restrictions are applied in an analysis of the WST in an effort to garner a set of possible inflectional suffixes. First, any suffix which has a suffix itself cannot be inflectional, based on the assumption that inflectional suffixes always occur at the end of a word. Second, root categories must have at least two inflected forms, thus a prefix may only be a possible root category if it appears to have at least two inflectional suffixes. The corollary is that a suffix is not inflectional if it is

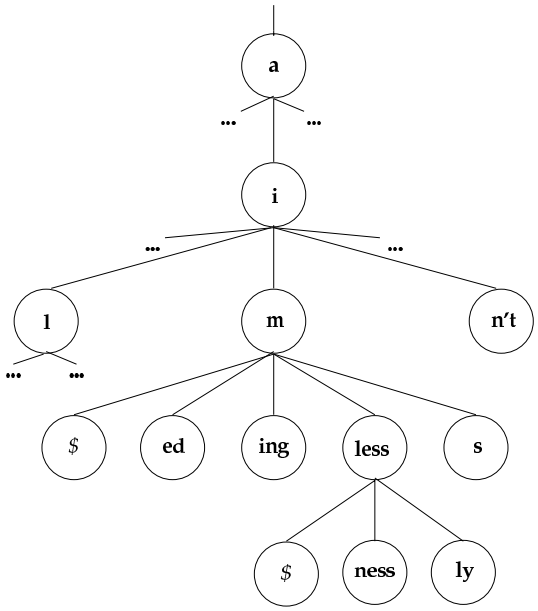

Figure 1: Portion of a word suffix tree.

the only inflectional suffix. Under these restrictions, the suffix set for "aim" in Figure 1 would be {*$* , *ed*, *ing*, *s*}.

When this technique is applied to an 80,000 word vocabulary of English[1], 1225 suffixes emerge as possible inflections. 1018 of these are affixed to just one word, only 42 are affixed to five words or more, and only 7 are applied to *at least* twenty-five words:

-s, -d, -ing, -ed, -ly, -r, -es

Under the assumption that inflection is a general grammatical process that applies to large numbers of lexical roots, we restrict the set of candidate inflectional suffixes to just these seven, plus the NULL suffix which marks uninflected forms. From a subjective analysis we recognize that *-d* is the same inflectional affix as *-ed*, but is a generalisation for verbs whose root ends with an 'e', such as *abate*. This proves unproblematic as it is eventually subsumed by the *-ed* suffix when feature dependencies are inferred later on. Similarly, the *-r* derivational suffix for nouns ending in an 'e' is subsumed by the more common *-$* for all singular nouns. The *-ly* adverbial marker does not go away so easily, but this is not a problem given that it is often useful to treat adverbs as inflected adjectives.

[1] taken from a machine-readable version of the Oxford English Dictionary, where only proper nouns and multiword entries were omitted.

Finding feature dependencies

To refine our notion of an inflectional suffix, and to accommodate irregularly inflected words, we generalise over regularities observed about the way crudely derived inflections are used. That is, words which share some or all of the same inflections, and are used in some or all of the same explicit contexts, are assumed to undergo the same inflectional processes and thus belong to the same lexical category. The analysis is restricted to trigrams because they are the shortest contexts able to capture the two regularities of interest: enclosing context, for generalising lexical categories, and prior context, for generalising projection.

Training is carried out on a large sample of text (such as the Brown Corpus) which is first processed so that each word inflected by one of the eight crudely derived suffixes is replaced by its suffix. The trigrams are abstracted out and processed in decreasing order according to their frequency. Processing is performed in two passes.

Categorisation

In the first pass, trigrams which differ only by the middle term are used to form broad lexical categories. Words which share at least one common inflectional suffix, and which appear in at least one common enclosing context, are assumed to belong to the same category and are consequently assigned a common part-of-speech label. The label is given as a feature whose value is specified by the most frequent inflectional suffix for the common context. For example, given the explicit sequences *the dogs bark*, *the cats purr* and *the foxes sleep*, suffix substitution would yield the following trigrams

the	-*s*	-*$*
the	-*s*	-*$*
the	-*es*	-*$*

The common enclosing context for the plural nouns is assumed to imply a common syntactic category for them. This is captured in the lexicon by associating a new category feature, *cat*, with each word, and the value of the feature is set according to the most frequent suffix as given by the common context, giving

$$w(cat = -s, \text{“}dogs'')$$
$$w(cat = -s, \text{“}cats'')$$
$$w(cat = -s, \text{“}foxes'')$$

Irregularly inflected words (such as “mice” for this context) inherit the common regular feature through subsumption.

As trigrams are processed, any set of words which includes some that are already categorised is analysed to see if its unlabeled words are inflectionally consistent with an existing category. If so, they are labeled with the existing part-of-speech tag; otherwise a new category is established. This helps mitigate the creation of an excessive number of overly-specialised categories.

Projection

In the second pass, trigrams which differ only by the last term are reconciled through the creation of additional features-values. As selection is generally rightward in English, the projection is assumed to stem from the middle term. For example, part of the ordered set of trigrams might be as follows:

the	-*s*	were
the	-*s*	-*$*
the	-*s*	-*ed*

Assume for this example that -*s* has replaced *dogs* as the second term, and that the prefix for -*$* and -*ed* is *walk* in the third position. These three trigrams imply a possible agreement constraint (in this case, number) that can be captured by a change to the feature structure. We assign an additional feature to the second word and instantiate it with the value of its suffix, giving

$$w(cat = -s, agr_1 = -s, \text{“}dogs'')$$

The projection principle thereafter allows us to attach the same feature-value pair to the third terms, giving

$$w(cat = -ed, agr_1 = -s, \text{“}were'')$$
$$w(cat = -ed, agr_1 = -s, \text{“}walks'')$$
$$w(cat = -ed, agr_1 = -s, \text{“}walked'')$$

Expressing the dependency

To express the agreement constraint as a grammatical dependency, a syntactic rule is created each time a new constraint is inferred. The rule expresses the linear precedence relation for a pair of terms, further constraining the relation by forcing feature agreement. For the example expressions given in the previous section, the creation of $agr_1 = -s$ is accompanied by the creation of a new unification rule

$$r(cat = X) \rightarrow w(cat = -s, agr_1 = X, A), \; w(cat = -ed, agr_1 = X, B).$$

This defines a grammatical sequence that consists of a word from category $-s$ followed by a word from category $-ed$ wherever such words agree with respect to their common feature agr_1. The rule itself inherits the unifying feature-value. It is envisioned that by substituting the left-hand rule symbol (along with its feature structure) into the text, further trigram analysis will permit creation of rules that account

for higher level structures allowing an iterative construction of a complete unification grammar. This is currently being studied.

3 Remarks

Unsupervised learning of unification grammars is of both practical and theoretical interest. To the extent that inflectional agreement morphology and syntactic agreement structures are linked, generalisation over inflectional suffixes is likely the only means by which this can be done from plain text alone. This research represents an initial attempt at doing just that, and highlights a number of incidental issues which should generate interesting discussion within a language learning workshop.

This paper shows that a WST is a suitable mechanism for uncovering morphemic suffixes, but in isolation is insufficient for identifying those which mark inflection. By generalising over the contexts in which individual suffixes appear, a reliable estimate of actual agreement constraints emerges, one which permits ready translation into unification rules for pairs of terms. Whether the iterative unification process described in the previous section turns out to be viable for constructing a complete grammar is presently being tested. It may turn out that statistical co-occurrence or techniques from probabilistic link grammars may be needed, particularly for capturing long-distance dependencies as these are unlikely to emerge within a simple trigram analysis. Results are expected to be available in time for the workshop.

References

Steven Abney. *The Noun Phrase in its Sentential Aspect*. PhD thesis, MIT, 1987. unpublished.

Steven Abney. Stochastic attribute-value grammars. *The Computation and Language E-Print Archive*, page 21, October 1996. 9610003.

A. Andersson, N. Jesper Larsson, and Kurt Swanson. Suffix trees on words. In D. Hirschberg and G. Myers, editors, *Lecture Notes in Computer Science 1075; Combinatorial Pattern Matching*, pages 102–115. Springer-Verlag, 1996.

Chris Brew. Stochastic hpsg. In *Proceedings of EACL-95*, 1995.

S. F. Chen. *Building probabilistic models for Natural Language*. PhD thesis, Harvard University, Cambridge, Massachusetts, Cambridge, Mass., 1996.

Andreas Eisele. Towards probabilistic extensions of constraint-based grammars. Deliverable r1.2.b, DYANA-2, September 1994.

Naoki Fukui and Peggy Speas. Specifiers and projection. *MIT Working Papers in Linguistics*, 8:128–172, 1986.

F. Jelinek, J. D. Lafferty, and R. L. Mercer. Basic methods of probabilistic context-free grammars. In *Speech Recognition and Understanding: Recent Advances, Trends and Applications. Proceedings of the NATO Advanced Study Institute*, pages 345–360, 1992.

Rick Kazman. Simulating the child's acquisition of the lexicon and syntax—experiences with *babel*. *Machine Learning*, 16:87–120, 1994.

A. Sanfilippo. Lkb encoding of lexical knowledge. In T. Briscoe, A. Copestake, and V. de Paiva, editors, *Default Inheritance within Unification-Based Approaches to the Lexicon*. Cambridge University Press, 1993.

Tony C. Smith and John G. Cleary. Probabilistic unification grammars. In *Workshop Notes: ACSC '97 Australasian Natural Language Processing Summer Workshop*, pages 25–32, Macquarie University, February 1997.

Hans Uszkoreit and Annie Zaenen. Grammar formalisms. In Ron Cole, editor, *A Survey of the State of the Art in Human Language Technology*, chapter 3.3. Center for Spoken Language Understanding, University of Pisa, Italy, 1996.