University of Pennsylvania Working Papers in Linguistics

Volume 12
Issue 1 *Proceedings of the 29th Annual Penn Linguistics Colloquium*

Article 4

2006

Unsupervised Morphology Induction for Part-of-speech-tagging

Damir Ćavar
Indiana University

Paul Rodrigues

Giancarlo Schrementi

Unsupervised Morphology Induction for Part-of-speech-tagging

Unsupervised Morphology Induction for Part-of-Speech Tagging

Damir Ćavar, Paul Rodrigues and Giancarlo Schrementi

1 Introduction

In this paper we discuss a specific approach and the role of unsupervised morphology induction for induction of lexical properties and part-of speech (PoS) tagging.

There is a clear intuition among native speakers that PoS classification of words depends on various factors, e. g. distributional properties or the words in context, as well as morphological structure of the particular tokens. Induction of word types was modeled in various approaches by mapping contextual and distributional properties (e. g. Mintz et al. 2002, Lee 1997) on vector space models and clustering on the basis of vector similarities.

Various PoS tagging algorithms make either use of manually coded contextual and morphological rules, or use learning and training approaches to exploit such information contained in large corpora via n-gram models and morphological classification, cf. Brants (2000), Lee et al. (2002). To take a particularly suggestive example, of the words in the WSJ section of Penn that end in "able", 98 percent are adjectives, and only 2 percent are nouns (e. g. "cable", "variable"; Brants 2000). This means that the suffix highly predicts the categorization of the word and is therefore a powerful aid to any PoS tagger.

Our suggestion involves unsupervised induction of morphological signatures for lexical items, on the one hand for pure lexicological purposes, i. e. for the study of induced lexical similarities and relations. On the other hand, the study of computational approaches to the extraction of concise grammars and sub-grammars for various linguistic levels from raw language data is important for not only the study of language learnability in general, but also of typological questions. A computational model for particular linguistic domains does not only offer potential solutions for less commonly studied and documented languages, it also helps deriving higher level linguistic knowledge.

In the following we show how a morphological system can be detected and induced from raw text, and how this knowledge can be used to derive not only lexical classification, but also morpho-syntactic knowledge for applications in the domain of PoS tagging.[1]

[1] The notion *unsupervised* seems to be a quite flexible term in the computational

2 Prior Work

Samuelson (1993) introduced an algorithm to utilize end-of-word substring or "suffixes" to categorize words into PoS by taking probabilities of substring word endings of 7 characters or less and smoothing them by averaging in the probability with one character less, with each iteration. This approach was combined in the TnT-tagger (Brants 2000) with a statistical n-gram model, where PoS of events that are not in the n-gram models are guessed via suffix sequence and PoS correlations trained on tagged corpora. Brants (2000) reports 89.0 percent accuracy on these unknown words using the Penn Treebank (Marcus et al. 1994) as a corpus.

Lee et al. (2002) performed a similar experiment on Korean. Their approach uses a morpheme pattern database to automatically tag the agglutinative morphology of Korean. After assigning all possible morpheme tags to a morpheme, the text is run through a statistical PoS tagger which uses the Viterbi algorithm to assign word categories. This is then run through a correction layer, using a rule-based correction system. Even though 10 percent of the words were unknown, Lee et al. report a tagging accuracy of 97 percent.

Specific precision and recall scores of the morphologic component alone in TnT were not reported. Lee et al. reported a 94.9 percent recall and 89.7 percent precision on the Korean data. In all these cases the significant aspect is that random suffix sequences that correlate with PoS in annotated corpora are used, or manually coded dictionaries and morphological rules. While automatically generated patterns focus on strategies that lack morphological insights, grammar based models tend to leak and miss statistical properties.

In our approach, which is similar to a more supervised strategy described in Goldsmith (2001), we concentrate on the question of whether a more general algorithm can be used to induce significant morphological cues for not only suffixing languages, but also for Semitic, agglutinative and polysynthetic languages, where the crucial morphological cue does not necessarily have to be right peripheral.

In the following we will describe our algorithm that uses variants of *Alignment Based Learning* (ABL) (e. g. Zaanen 2001), *Longest Common Subsequence* (LCS), and an *Interdigitation* and *Layer*-based string analysis for hypothesis generation on the morphological level. In previous studies (e. g. Ćavar

linguistic literature. Our understanding of unsupervised systems is maybe more restrictive, i. e. stating that no language specific knowledge is involved, except for maybe the information that natural language utterances are sequences of non-discrete events that are mapped on discrete symbolic sequences.

et al. 2004b,a) we presented versions of this morphology induction algorithm for the induction of a lexicon and morphological rules for a wide range of natural languages. The resulting morphological rules and structures were optimized during the induction process using a constraint satisfaction model which enforces preferences as to the size and statistical properties of the respective grammars. In particular, we used constraints based on *Minimum Description Length* (MDL), *Relative Entropy* (RE) or *Kullback-Leibler Divergence* (KLD), and *Maximum Average Mutual Information* (MI). Tested on various languages and different types of corpora[2], the resulting morphological segmentation reached approx. 99 percent precision over all languages to varying levels of recall.

Given the very precise morphological grammar we were able to generate, lexical classification was performed on the basis of the resulting signatures together with distributional context vectors with soft clustering algorithms, resulting in separation of the elements into basic induced lexical classes that are mapped via human evaluation on deductive tags, e. g. verbs and nouns. Our algorithm's high precision and lower level of reliance on supervised knowledge makes it an attractive replacement for either of the mentioned approaches that rely on morphological cues for lexical typing. In the following, we will present a new approach of the morphology induction algorithm and new experiments in the domain of category guessing on standard corpora.

3 A Constraint Satisfaction Model

The morphological induction is centered around two components. The first, which we will call GEN, generates hypotheses for possible morphological segmentations of a word. The second, EVAL, evaluates the hypotheses and ultimately selects what it considers the best one. These two components are coupled with a memory subsystem that consists of a long-term and a short-term memory. The long-term memory is the accumulated knowledge of the system, containing information about known morphemes and their n-gram contexts. The short-term memory consists of recent segmentations that are still up for revision in the near future. Its main purpose is to allow the program to see potential segmentations that might be more optimal over a range of several utterances.[3]

[2] We used literature, newspaper articles, and child-oriented speech from CHILDES corpora.

[3] Without the short-term memory component it would be almost impossible to generate segmentation hypotheses for languages with extremely low token frequencies,

The algorithm assumes as input a sequence of utterances with existing word boundaries marked with any number of white-space characters. It proceeds incrementally through each word, generating hypotheses for that word's segmentation and then evaluating them on the basis of prior knowledge and statistical properties of the segments. The selected hypothesis optimizes the grammar memory and distributional criteria. This results in the best hypothesis being incorporated into the knowledge of the long-term memory and the new hypothesis being added to the short-term memory of recent segmentations. The following pseudo code describes the general incremental loop of the algorithm:

```
WHILE input-utterance:
   FOR-EACH-WORD:
      generate segmentation hypotheses
      evaluate segmentation hypotheses
      add-hypotheses to short-term memory
      add optimal hypothesis to long-term memory
```

For memory and grammar calculations we use three different data structures:

- morpheme hash tables with frequency counts
- bigram hash tables with frequency counts
- multigram hash tables with frequency counts

The multigram hash tables store an ordered list of morphemes in a learned segmentation and thus represent the complete word as a concatenation of the sub-morphemes. The length of the multigrams can vary from 1 to n.

4 Gen

ABL and similar hypothesis generation strategies have the advantage of being completely memory driven. Morpheme boundaries are assumed at the alignment positions between two strings, where one is the input word, and the other one is a word from memory. In principle, such a strategy reduces the number of possibilities for segmentations, compared to the total explosion of one word into all possible segmentations. However, a growing lexicon of morphemes

i. e. agglutinative and polysynthetic languages.

leads to many substring comparisons, such that even a optimal data structure strategy cannot reduce a prohibitively large number of matching computations. We decided to apply alignment tests by generating all possible segmentations of a word and checking whether the segments exist as morphemes in memory.

This strategy turns out to be efficient enough for suffixing languages, e. g. Indo-European languages. For languages with very low token frequencies, like for example Japanese, a large number of tokens would be needed to identify common subsequences, e. g. suffix particles or postpositions. In order to cope with such problems we used a Longest Common Subsequence (LCS) algorithm. However, the most optimal LCS algorithm we are aware of still requires O(m+n) * MORPHEMECOUNT steps (Freschi and Bogliolo 2004). Thus, for efficiency reasons we restricted the LCS calculations to word edges only, i. e. searching at the left and right periphery of the new word, and comparing only with the words stored in short-term memory. Nevertheless, additional hypothesis generation methods are necessary to identify interdigitation between roots and vowel layers in Semitic languages. Some possible solutions were suggested in (Rodrigues and Ćavar 2005).

We classify morphemes into four groups: independent, left-independent, right-independent and dependent. Independent morphemes need not to have another morpheme to their left or right in a segmentation. Left-independent don't need a morpheme to occur to their left but must have one to their right. Right-independent are the reverse of left-independent and dependent morphemes must have morphemes on both sides of them. This categorization scheme allows for morphemes that have the same character string but different morphological roles to keep their distinction in the knowledge of our system. It also has the benefit of being able to be derived from the data without any sort of prior knowledge other than to be aware of these lateral relationships.

4.1 Evaluation of Hypotheses

EVAL uses a voting architecture based upon a series of metrics to evaluate the hypotheses. The hypotheses are ranked with regards to each metric and their rankings are added up to produce a final score for that hypothesis. The hypothesis with lowest score value is then judged to be the best hypothesis. For example, a hypothesis that managed to be first in each of the eight metrics would have a final score of 8, whereas one that was second in each of the metrics would have a score of 16.

For the metric we used two types of constraints:

- Memory-oriented constraints

- Processing-oriented constraints

Memory-oriented constraints favor compression of the grammar and language data, in the sense of the MINIMUM DESCRIPTION LENGTH PRINCIPLE (Grünwald 1996, Grünwald et al. 2005), while processing-oriented constraints favor less complex segmentations and faster access and generation.

Since all our data-structures are probability distributions of n-grams and multigrams, we reduce these constraints to basic Information Theory relations, based on the notion of entropy.

4.1.1 Metrics

One of the central constraints for minimization of the grammar size is MINIMIZE KULLBACK–LEIBLER DIVERGENCE (KLD). KLD favors the hypothesis that leads to the smallest increase of memory size, using the following formula over the morpheme, bigram, and multigram distributions:

$$d = \begin{cases} \sum_{x \in H} \left(p(x) lg \left(\frac{p(x)}{q(x)} \right) \right) & \text{if } x \in q(x) \\ \\ \sum_{x \in H} \left(p(x) lg \frac{1}{p(x)} \right) & \text{if } x \notin q(x) \end{cases} \tag{1}$$

KLD compares for each hypothesis the number of bits needed to add the new hypothesis to the existing probability functions. If a morpheme or n-gram is not found in memory, the costs are assumed to be the entropy of the new outcomes, assuming a *what-if* calculation, where the relative frequencies of the elements in the existing grammar are reduced by the probability of the new event. It is important to realize that the calculation of KLD in our incremental learning model is restricted to the morphemes and n-grams in the hypotheses only. We do not recalculate the size of the complete *grammar* every time a new hypothesis is added, but rather estimate the additional costs of adding a hypothesis to the model. This restriction reduced the necessary calculations and comparisons to the necessary minimum.

A further metric is related to the likelihood that two morphemes represent a sequence of a word. We assume that this is high if the Mutual Information that one morpheme contains about another is high. The constraint MAXIMIZE MUTUAL INFORMATION expresses this intuition. It calculates the number of bits that could be spared if two morphemes are stored together, rather than as separate elements in a language model. The following formula shows how we calculate this constraint over all bigrams in a given hypothesis, assuming several competing hypotheses as for the possible segmentations of *sleeps*:

- Input: *sleeps* H_1: *sleep s* H_2: *s leeps*

$$I(sleep; s) = P(sleep, s) lg \frac{P(sleep, s)}{P(sleep)P(s)} \tag{2}$$

To better capture relations and compressions between the uni- and bigram models, we decided to take a variant of KLD and call it MINIMIZE RELATIVE ENTROPY (RE), where the comparison between the two distributions takes conditional probability of elements in bigram sequences into account. We calculate RE as follows:

$$d = \sum_{x \in H} \sum_{y \in H} \left(p(y) lg \left(\frac{p(y)}{p(y|x)} \right) \right) \tag{3}$$

Again, the number of calculations is restricted to the number of morphemes and bigrams in the set of hypotheses for a given new word.

Various other minor constraints are taken into account, favoring more or less segmentations, favoring longer or shorter morphemes, segmentation lists, and so on. We will not go deeper into these constraints, since in principle, the three major constraints mentioned above turned out to be fully sufficient for the majority of languages we evaluated.

Each of these constraints establishes a ranking table, with the most favorable hypothesis getting the best voting, expressed numerically.

The hypothesis with the highest number of votes from all constraints is considered the winner and enters memory. All other hypotheses are remembered for a determined number of subsequent input sequences and used in hypothesis generation and evaluation as described above.

In initial experiments we discovered that these constraints tend to play different roles in different languages. In order to provide more dynamics in the learning phase and more self-adaptability for different types of languages, we weighted all constraints, thus relativizing the resulting votes and providing means for more flexibility. The problem, however, due to our decision of no supervision in the system, was to provide means for self-supervision and automatic adaption of the weights for each constraint. We did not evaluate different self-supervision strategies yet, but so far, our impression is that an error driven and time-based weight adaption might lead to the best results. One of the basic problems with the quantitative constraints we use is low frequency of morphemes in the initial phase. This leads to high scores from MI, due to its properties. A continuous increase of the MI weight leads to better results during the growth of the language model. Due to place restrictions, we cannot go into details here, but only mention that a language model size dependent weight of the probability-based constraints seems most effective with the least computational effort.

5 Category Induction

The learning model is incrementally generating language models or even morphological grammars that are subject to dynamic change with every new input. This model allows us to study how potentially frequency dependent effects might emerge, e. g. the phenomenon of apparent learning phases. From the language acquisition literature and personal communication with many researchers in this domain we found out that the acquisition of morphology is subject to phases. In English, for example, it seems to be the case that children first acquire nominal and verbal suffixes, and in particular inflectional suffixes. Derivational morphology, as well as possible prefixes seem to be acquired subsequently. One possible explanation for this observation might be the frequency of these morpheme types. We observed in manual segmentation and calculations that inflectional morphemes (e. g. *-ing*, *-s*, *-ed*) are twice as frequent as other morpheme types. While this might be an accident in our experiments, we observed that these morphemes are the first ones that appear in the segmentations of words. Since our model is extremely frequency dependent, we predict such developmental phases to show up in other languages as well, as long as they are correlated with frequency profiles of the respective morphemes.

The resulting language model is not only dynamic and can be saved any time in the learning process, and studied for any input phase, it also contains purely descriptive information about the morphemes in their context. One possible use of this information might be in the domain of category induction. While the above mentioned literature discusses the role of distributional properties for lexical typing, so far the role of morphological cues for lexical type induction was not dominant, or as in the example of the TnT tagger restricted to simple right peripheral character sequences, rather than real morphemes.

In subsequent experiments we used the morpheme collocation patterns to generate a vector space and test category induction with the use of morphological cues alone, and together with distributional cues, i. e. words in the local context (one word to the right and left). As already mentioned above, clustering studies have shown that distributional properties are potentially good cues for the differentiation of word types in English (and similar languages). Morphological cues are expected to boost this effect even further. The morphological collocation patterns that we can generate are of the following form:[4]

```
show (51 3 (((_@0) 48)
            ((_@0 ing$) 1)
            ((_@0 s$) 1)
            ((_@0 ed$) 1)))

man  (55 3 (((_@0) 48)
            ((#train$ _@0) 1)
            ((_@0 's$) 5)
            ((#horse$ _@0) 1)))
recorder (48 3 (((_@0) 46)
                ((_@0 's$) 1)
                ((_@0 s$) 1)))
Lois (59 3 (((_@0) 57)
            ((_@0 'll$) 1)
            ((_@0 's$) 1)))
ed  (21 2 (((#want$ _@0) 1) ((#roll$ _@0) 1)
           ((#turn$ _@0) 1) ((#push$ _@0) 1)
           ((#open$ _@0) 1) ((#us$ _@0) 1)
           ((#pick$ _@0) 1) ((#pour$ _@0) 1)
           ((#need$ _@0) 1) ((#jump$ _@0) 1)
           ((#start$ _@0) 1) ((#fill _@0) 1)
           ((#learn$ _@0) 1) ((#crack$ _@0) 1)
           ((#dump$ _@0) 1) ((#ask$ _@0) 1)
           ((#stuff$ _@0) 1) ((#call$ _@0) 1)
           ((#miss$ _@0) 1) ((#show$ _@0) 1)
           ((#hand$ _@0) 1)))
```

On the one hand, it is immediately clear how one can use the signatures as such for lexical type induction. Just the length of the signature, the type of co-occurrence morphemes, and their frequency provides enough obvious information for lexical classification. On the other hand, this information together with distributional properties should enhance lexical type identification

[4] These signatures were generated from the child-oriented speech in the Peter corpus (Bloom, 1970) in CHILDES. The initial number in the bracketed structure is the total morpheme count. The second number represents an internal ID which signals whether the morpheme was seen independent of other morphemes, left-, right-, or both-sides-dependent. The sequence _@0 is a place holder for the morpheme, possible contextual morphemes are listed, and the sequence as such is counted.

even more, given that certain co-occurrences on the token level are extremely significant, like for example the sequences "*the* + NOUN" and "*a* + NOUN". Although we integrated vectorization and clustering algorithms in this system, we will focus in the following on the discussion of the fundamental cue-induction results.

6 Results

On various types of corpora from one language the system performs differently. Experiments on CHILDES corpora have surprisingly shown very good results. The annotations and transcriptions in CHILDES corpora vary dramatically. Many spoken language phenomena are integrated into the transcription schema, such as cliticization and fusion phenomena, which make morphological segmentation difficult even for human evaluators. On the other hand, such corpora have a very different type-token ratio than e. g. newspaper articles, i. e. a few tokens are used over long passages in different contexts many times. Thus, our algorithm performs best on spoken language transcripts, and interestingly enough, best on child oriented speech.

For the Peter corpus the number of errors is limited to three segmentations, all other segmentations were accurate for a human evaluator. Some problems are due to mismatches between orthography and pronunciation, e. g.:

```
let (92 3 (((_@0) 91) ((_@0 ting$) 1)))
```

Overall, the segmentation achieves 99 percent precision on English corpora, with a recall in the range of 80 percent. The evaluation of the recall is in particular very difficult, due to the lack of appropriate corpora. We based various evaluations on an automatic comparison with the segmentations found in the CELEX database, as well as on manually segmented word lists.

As expected, the performance of an ABL-based hypothesis generation on agglutinative languages is extremely bad. The amount of necessary input is very high, to result in basic morpheme signatures, given a segmentation strategy that requires the existence of sub-morphemes in memory. Further evaluations will show how the LCS-based approach with varying short-term memory size performs on such languages.

On the other hand, the extremely good results for Indo-European type of suffixing languages shows two things:

- It is possible to identify morphological cues with high precision for higher level grammar induction.

- The required computational effort is relatively small, and increases with other language types (i. e. agglutinative and polysynthetic languages).

The prediction is thus, the role of morphological cues will be different in other language types, and the amount of input data necessary to identify basic morphological cues will vary dramatically across languages.

However, for English, these cues are easy to identify. But, in order to be able to appreciate this finding, we need to identify their potential role in lexical typing. It is important to see what the base-line contribution of such morphological cues for PoS could be.

In order to establish the base-line, we transformed the morphological signatures gained from the analysis of child-oriented speech into regular expressions. We found basically two different signature types in the first phase (initial 4 documents of the Peter corpus, with the following properties: utterances: 12326; tokens: 43646; types: 1583; bigram tokens: 31320; bigram types: 8533):

```
(ing|s|ed) - for V
('s|'ll|s) - for N
```

The question now is, how much information can these cues contribute to knowledge about lexical types?

We used the Brown corpus to evaluate the simple task:

- Replace all matching words with the tag V or N for the two patterns and calculate the precision and recall score.

For each morpheme that was identified, we calculated the correspondence with the reduced Brown-tag as given in the following table:

write V when *s	
nouns	56%
verbs	23%
write V when *ing	
nouns	20%
verbs	70%
write V when *ed	
nouns	2%
verbs	92%

This shows that only some of the morphemes contribute a lot to the specificity of the lexical type in this case. We can probably generalize, without having tested this, to all single morphemes. On the other hand, the signature as such is probably highly significant for the lexical type, i. e. the pattern

(WITHOUT MORPHEME, WITH S, WITH ING, WITH ED) is highly significant as a clue for verbs in English.

In comparison, we added the most significant token in the bigram model as additional contextual information and performed the same task on the Brown corpus for nouns:

write N when **the—a *s**	
nouns	90%
verbs	1%
write N when **the—a *'s**	
nouns	99%
verbs	0%
write N when **the—a *'ll**	
nouns	100%

This comparison shows clearly that these few morphemes in combination with the most frequent token co-occurrence pattern derive extremely reliable lexical type information.

On the other hand, it is clear that for a general lexical typing the signatures as a whole are as crucial for lexical typing, not only due to morphological ambiguity, but also due to recurrent patterns in other types of lexical forms.

We have shown that with a quite simplistic computational architecture it is possible to induce morphemes with a high accuracy. The complexity of the computational means is related to the complexity of the linguistic level of morphology for each language. We expect this task to be more complex for languages with high type and low token frequencies. Further, we have shown that in English very simple collocation patterns together with basic morphemes derives highly accurate type information for the two most basic lexical classes, without reference to higher level grammar rules or large and specific lexical knowledge.[5]

References

Brants, Thorsten. 2000. TnT – A statistical part-of-speech tagger. In *Proceedings of the 6th Applied NLP Conference, ANLP-2000*. Seattle, WA.

Ćavar, D., J. Herring, T. Ikuta, P. Rodrigues, and G. Schrementi. 2004a. Alignment based induction of morphology grammar and its role for bootstrapping. In *Proceed-*

[5] As a final remark, we would like to add that the source code (written in Scheme) together with evaluation samples on different types of languages is available for related research projects and experiments. Please contact the authors, if you are interested in evaluations on other languages or cross-studies.

ings of Formal Grammar 2004, ed. G. Jäger, P. Monachesi, G. Penn, and S. Winter, 47–62. Nancy, France.

Ćavar, D., J. Herring, T. Ikuta, P. Rodrigues, and G. Schrementi. 2004b. On statistical bootstrapping. In *Proceedings of the First Workshop on Psycho-computational Models of Human Language Acquisition*, ed. W. G. Sakas, 9–16. Geneva, Switzerland. Held in cooperation with the COLING 2004.

Freschi, Valerio, and Alessandro Bogliolo. 2004. Longest common subsequence between run-length-encoded strings: a new algorithm with improved parallelism. *Information Processing Letters* 90:167–173.

Goldsmith, John. 2001. Unsupervised learning of the morphology of a natural language. *Computational Linguistics* 27:153–198.

Grünwald, Peter. 1996. A minimum description length approach to grammar inference. In *Symbolic, Connectionist and Statistical Approaches to Learning for Natural Language Processing*, ed. G. Scheler S. Wermter, E. Riloff, volume 1040 of *Lectur Notes in Artificial Intelligence*, 203–216. Berlin: Springer Verlag.

Grünwald, Peter D., In Jae Myung, and Mark A. Pitt, ed. 2005. *Advances in Minimum Description Length: Theory and Applications*. Cambridge, MA: The MIT Press.

Lee, Gary Geunbae, Jeongwon Cha, and Jong-Hyeok Lee. 2002. Syllable-pattern-based unknown-morpheme segmentation and estimation for hybrid part-of-speech tagging of Korean. *Computational Linguistics* 28:53–70.

Lee, Lillian. 1997. Similarity-Based Approaches to Natural Language Processing. Doctoral dissertation, Harvard University.

Marcus, Mitchell P., Beatrice Santorini, and Mary Ann Marcinkiewicz. 1994. Building a large annotated corpus of English: The Penn Treebank. *Computational Linguistics* 19:313–330.

Mintz, Toben H., Elissa L. Newport, and Thomas G. Bever. 2002. The distributional structure of grammatical categories in speech to young children. *Cognitive Science* 26:393–424.

Rodrigues, Paul, and Damir Ćavar. 2005. Learning Arabic morphology using information theory. In *Proceedings of the 41st Annual Meeting of the Chicago Linguistics Society (CLS 41)*. Chicago, IL.

Samuelson, Christer. 1993. Morphological tagging based entirely on Bayesian inference. In *9th Nordic Conference on Computational Linguistics NODALITA-93*. Stockholm, Sweden: Stockholm University.

Zaanen, Menno M. van. 2001. Bootstrapping Structure into Language: Alignment-Based Learning. Doctoral dissertation, The University of Leeds, Leeds.

Department of Linguistics
Indiana University
Bloomington, IN 47405
dcavar@indiana.edu