An Architectural Account of Errors in Foreign Language Learning

Julie VanDyke and **Jill Fain Lehman**
Carnegie Mellon University
Computer Science Department
Pittsburgh, PA 15213
{vandyke,jef}@cs.cmu.edu

Abstract

It has often been observed among teachers of English as a foreign language that the English article system is difficult for learners to master. This paper provides a processing account which pinpoints the source of these errors as being within the learner's architecture for production. We illustrate our account with a computational model of one group of foreign language learners embedded within NL-Soar. The model's control structure and learning mechanism are used to explain the architectural character of errors and to predict the conditions required for overcoming them.

Learning problem

Teachers of English as a foreign language will generally agree that English articles are among the most resistant forms to master. Additionally, it appears that the degree to which a learner is able to master these forms is correlated with the presence or absence of articles in the learner's native language (Tarone and Parrish, 1988; Ringbom, 1985; Duskova, 1969). One popular explanation for this learning problem is that the semantics of the articles is unknown, unlike plural or tense markings where learners already know the semantics and simply have to learn new linguistic mapping rules. This paper describes a preliminary investigation of this issue from the perspective of the Polish native speaker. Polish is an appropriate choice because, like other Slavic languages, it does not have linguistic forms corresponding to English articles. We attempt to show that although Polish does not have articles like English, Polish learners do at least know the semantic features which they encode. Consequently, the problem is not a simply matter of semantics. Instead, we illustrate that the source of the problem lies in what we call the learner's **Production Architecture**. The production architecture is the particular organization of linguistic knowledge that is the product of native language acquisition.[1] We show that the reason Polish learners have such difficulty with English is because there is a *mismatch* between the native production architecture which the learner brings to the task of learning the foreign language and the architecture required for production in that language.

[1] This production architecture is language-specific, and should be thought of as distinct from the underlying general cognitive architecture.

The semantics of English articles

In light of the ease and frequency with which native speakers use articles, it may be somewhat surprising to discover the vast complexity of the system that learners must master. Leaving aside issues relating to the learner himself, (i.e. his level of proficiency and the potential for transfer and/or interference from his native language), (Young, 1996) describes four features which are necessary for the complete specification of article contexts: countability, number, reference, and discourse marking. The role of countability and number is illustrated in (1) below, borrowed from (Young, 1996):

```
(1) Piaget saw the growth of language  as
    tied to the growth of thought, as though
    it were a branch on the cognitive plant.
    Chomsky is inclined to see language
    learning and cognitive development as
    independent plants in a common garden.
```

Notice that in order for a learner to produce these sentences, he would have to recognize that the indefinite article *a(n)* should not be used with the noncount nouns *language* or *thought*, even though in other contexts it would be necessary:

```
(2) *I have just had thought.
```

Moreover, the user would also need to recognize that the zero article should be used with the plural count noun *plants* as in *independent plants*, but in other situations it will require an article:

```
(3) *I forgot to water plants in my window.
```

To completely specify these article contexts the learner must further incorporate the reference and discourse status of each NP into his application rule for each article type. Reference is characterized as either specific [+SR] or non-specific [-SR]; discourse status is characterized as either known to the hearer [+HK] or unknown to the hearer [-HK]. Thus [+SR,+HK] contexts would be marked with *the* and [+SR,-HK] contexts would be marked with *a(n)* or *0* depending on the countability of the noun. The [-SR] contexts have a less straight forward mapping, as they correspond to generic and attributive uses which can be marked by any article in English.

Linguistic Universals Of our four features for article specification, we will take the universality of countability and number for granted. Instead, we focus our discussion on the abstract features of [SR] and [HK], which

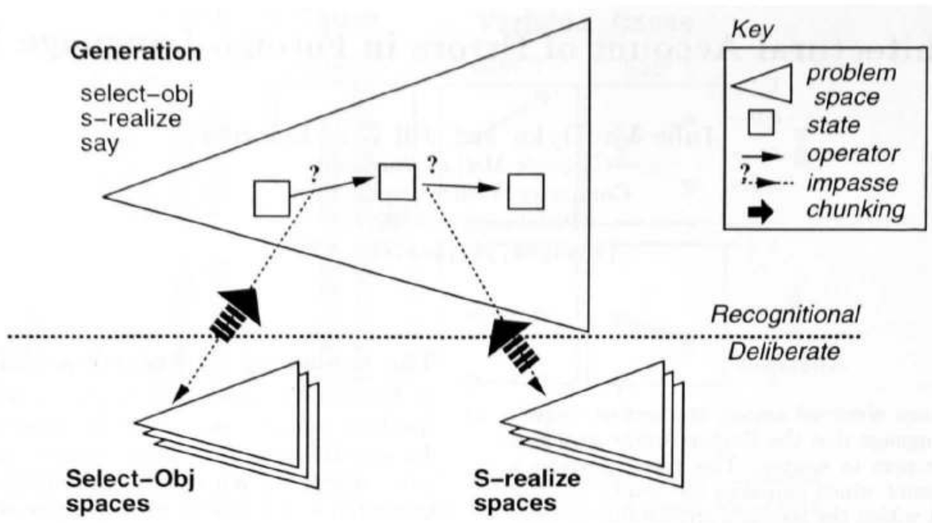

Figure 1: Processing in NL-Soar

were originally proposed by Bickerton as semantic and discourse universals for characterizing noun phrase reference (Bickerton, 1981). The claim is that these concepts are intrinsic to successful linguistic communication, but that different languages will realize these meanings differently. A number of studies have given support to this viewpoint (Huebner, 1983; Tarone and Parrish, 1988; Chauldron and Parker, 1990). The results suggest that learners progress through a series of stages, called *Interlanguage*, in which they distinguish among various feature structures which may or may not include the feature sets relevant to English. Consequently, the rules they create using these "intermediate" feature structures are not necessarily congruent with the rules of English, leading to production errors.

The immediate question we must ask then, is how does the Polish language capture these universals? Just as with the other Slavic languages, Polish captures discourse status [HK] with word order (Szwedek, 1976; Duskova, 1969). Thus, the sentence in (4) suggests that *chlopiec* is a new discourse referent (i.e. [-HK]) because it appears in clause-final position.

```
(4) Wszedl           chlopiec.
    go-past-3p-masc boy-nom-sg
    [A boy went out.]

(5) Chlopiec   wszedl.
    boy-nom-sg go-past-3p-masc
    [The boy went out.]
```

If the order were switched as in (5), *chlopiec* would be taken to refer to a discourse entity already introduced in the discourse, and hence already known to the hearer (i.e. [+HK]).

The [SR] universal is evidenced in Slavic languages through verbal aspect (James, 1969). [2] Thus in (6) the imperfective verb suggests an activity that is in progress; the salient aspect is the reading and no particular book is assumed. In (7) however, the book is assumed to have a particular referent.

```
(6) Ja czytalem          ksiazke.
    I  read-imperfective book-sing
(7) Ja przyczytalem  ksiazke.
    I  read-perfective  the book-sing
```

Thus we see that these semantic concepts are not unknown to the Polish native speaker and it is therefore insufficient to claim lack of knowledge as the source of learning difficulties. We will now turn to our process model for more insight into what makes articles so hard for Polish learners of English to master.

The Model

Role of Knowledge

Our approach to modelling a learner of English as a foreign language is fundamentally knowledge-based. The key questions we ask in identifying the source of errors are 1) Does the learner recognize the semantic concepts encoded in the forms of the second language, 2) Is this knowledge in the appropriate form for application, and 3) Is the learner able to use this knowledge at the appropriate times during processing? The answer to the first question has been addressed by our above discussion of linguistic universals; the learner has the knowledge of the relevant semantic concepts. As we will see below, the real source of difficulties lies in the form (question 2) and application (question 3) of that knowledge. To approach these questions, we have built a model of a Polish native speaker and set it to the task of generating English sentences requiring articles. The model is an

[2] We can not say that the correspondence to [+/- SR] is one-to-one. Nevertheless, there is a correlation sufficient to show that speakers do have access to this semantic feature.

elaboration of NL-Soar (Lewis and Lehman, 1997; Lewis, 1996) which is a cognitive model of real-time language comprehension and generation embedded within a single agent.

Knowledge Compilation in NL-Soar

In NL-Soar, as in the general Soar cognitive architecture, linguistic behaviour is modelled as a series of transitions between states. This is illustrated in Figure 1 with each state being represented as a small square. Each transition is brought about through the application of an *operator*, represented as arrows between states.[3] Operators are NL-Soar's primitive unit of work and the level at which the system makes decisions about what to do next. Behaviour is *recognitional* if NL-Soar knows what to do at every state in order to move to the next state. If NL-Soar doesn't know what to do next, it means that it has not learned the appropriate operator to transition out of the current state. In response to this lack of knowledge, the architecture automatically creates an *impasse* and NL-Soar goes into a *deliberate*[4] processing mode in which it searches its linguistic problem spaces for knowledge about what to do next. Each problem space contains syntactic, semantic, or pragmatic knowledge about the kinds of structures that lead to a well-formed utterance and situation model. When a solution is found, the architecture automatically returns the result as a *chunked* operator which integrates the information from all the knowledge sources used to produce the result. The next time the state that led to the impasse is encountered, rather than deliberating, the chunked operator will fire and processing will continue recognitionally. Thus, the role of Soar's chunking mechanism is to capture the conditions that led to the impasse and then compile the knowledge used to resolve it into an IF-THEN rule.

Knowledge Types

The operators discussed above are just one type of knowledge necessary for modeling intelligent cognitive behaviour in NL-Soar. Consider the situation in Figure 2 where there are several different operators which could apply to transition from a given state. Since NL-Soar can execute only one operator at a time, and after that operator has executed the system is in a new state, knowledge is necessary to choose among alternative operators. This knowledge is referred to as *search control* and embodies the information needed to guide processing along a single path to the desired goal.

Sentence Production

The starting point for sentence production in any language is conceptual structure. To set up the task in NL-Soar we begin with a semantic representation of the sentence to be generated.[5] For example, to generate the

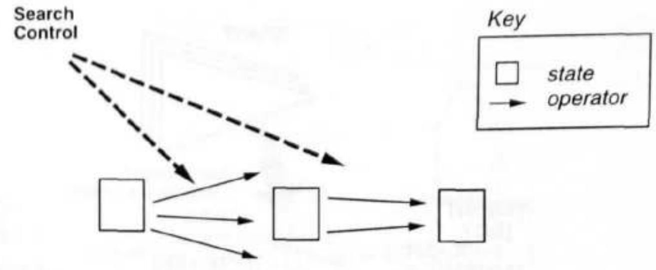

Figure 2: Search Control Knowledge

sentence **The boy left**, NL-Soar starts from the following:

```
[ACTION (go [AGENT ([ENTITY (boy (HK,SR))])]
      [PATH (from ([PLACE (cafe (HK,SR))]))])]
```

English Generation

What to say? To produce this sentence in English, first NL-Soar uses knowledge about which semantic concepts should be communicated to decide **what** it needs to lexicalize. This knowledge can be thought of as a combination of pragmatic knowledge about the discourse and knowledge of English grammar. Thus, in our example the system knows it must lexicalize the subject (i.e. [ENTITY (boy)]) and the verb (i.e. [ACTION (leave)] since English does not permit leaving either of these out in this context. The system may, however, choose not to produce a word for the embedded concept of PLACE because it knows that this is shared information from the discourse. Crucially, its knowledge of English will also tell the system to lexicalize the concept (HK,SR). We can now count three semantic concepts which the system as selected for lexicalization. In NL-Soar terms, we say that the system compiles all the knowledge it used to make these decisions into *chunked* operators called **Select-Obj** and consequently we see three Select-Obj operators in Figure 3.

When to say it? Having applied Select-Obj operators to the full semantic-conceptual structure to be communicated we have picked out three objects that must be lexicalized. The system now must make a decision about which one to say first. In NL-Soar terms, this knowledge is *search control*, the content being knowledge of English word order. The system will look at the three Select-Obj operators and choose [HK,SR] to be lexicalized first.

How to say it? Having chosen which semantic concept will be lexicalized first, there is still an open question about what English word to use. This is the function of NL-Soar's *S-Realize* operator. The operator is constructed by compiling knowledge from the S-Realize

[3] There are 3 classes of operators in our Generation model: we will use the labels *Select-Obj, S-Realize, Say* as mnemonics for the type of work they do.

[4] We do not mean to imply that this is conscious or intentional processing, only that it is not automatic.

[5] In fact we begin with the situation model NL-Soar builds after comprehending the Polish sentence whose English translation we desire to generate. In this way we constrain our semantic representation to the same representation used during comprehension, shown here in uppercase letters. Since the topic of this paper is one of errors in sentence *production* we refer the reader to (Lewis and Lehman, 1997) for a fuller explication of NL-Soar comprehension.

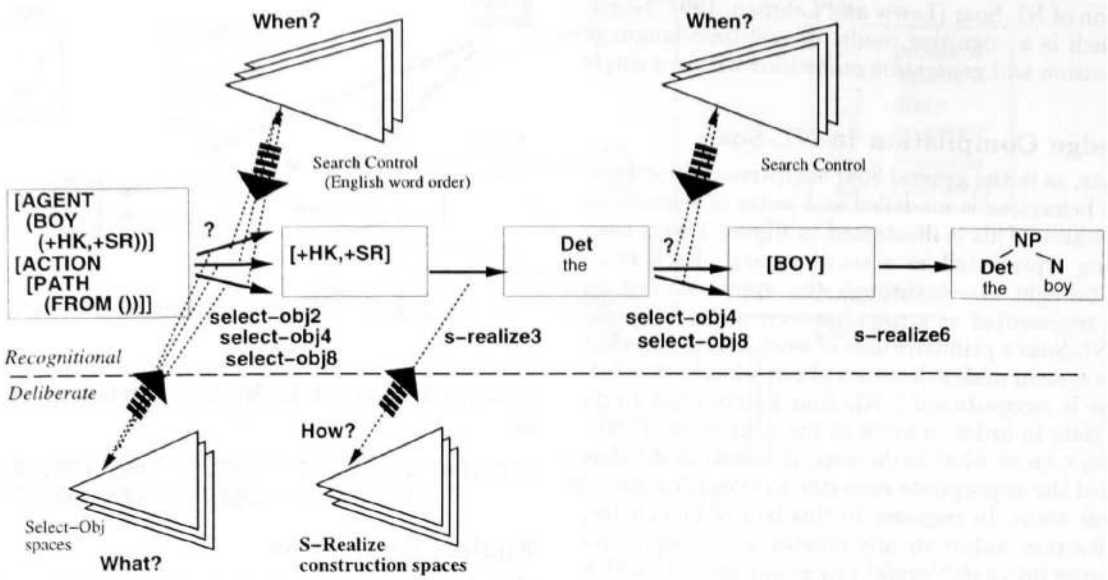

Figure 3: English generation of *the boy*

spaces that may include choices between synonyms or competing lexical forms. Here is where the system decides to use *the* for [HK,SR] or a pronoun like *this* or *that* that would communicate the same semantics. In the example in Figure 3 we see that the the S-realize operator chooses the word *the*.

NL-Soar's generation continues through this series of **what-when-how** decisions until all the Select-Obj operators have been realized. For fully learned behaviour, the system does not have to impasse into the problem spaces below the *Recognitional/Deliberate* dotted line because all operators will already be chunked. These decisions and the problem spaces used to create collections of chunked operators make up our model of the native English production architecture.

Polish Generation

What to say? We now turn to the architecture for production that underlies Polish generation, illustrated in Figure 4. As we have said previously, Polish uses word order and verb aspect to express the information contained in English articles. Thus when NL-Soar generates our sentence in Polish, there are only 2 Select-Obj operators, one for the subject and one for the verb. Unlike in the English case, where [HK,SR] was identified as a separate concept to be lexicalized, in Polish it remains part of the semantics of the subject.

When (and how) to say it? In Polish the semantics for definiteness, [HK], doesn't play a role until the system is trying to decide between the two Select-Obj operators. Here, it will notice that the subject is marked as [+HK] (i.e. known to the hearer) and recall that Polish grammar puts old information in sentence final position. Hence the Select-Obj operator corresponding to the verb will be chosen to execute first and an S-Realize operator will fire to determine how to say the verb, producing a lexicalization which takes the [SR] feature into account.

Architectural Mismatch

Now we can begin to see the difference in the architectures required for producing English vs. Polish sentences. For English, the [HK,SR] feature becomes a Select-Obj operator to be realized lexically from the very beginning of the production task; it is part of the **what** decision. Having selected the concepts to be realized, English word order is used to guide the sequence of production from then on. In Polish, however,[HK] information is the root of the **when** decision and [SR] information is part of the **how** decision. Rather than being an operator, as in the English case, the [HK] information is embodied in search control knowledge used to discriminate between operators. Thus both the organization of knowledge, and the time course for application is different across the two languages.

Explaining Production Errors

Incorrect use of articles distinguishes the English of even advanced learners of English as a foreign language. The Interlanguage literature (Larson-Freeman, 1975; Ringbom, 1985; Duskova, 1969) suggests learning stages which include 1) periods of using no articles; 2) periods of overgeneralization of *the*; and 3) periods where the learner has more discriminating Interlanguage rules whose closeness to Standard English varies across individuals. A learner may go through any or all of these stages and each stage may last for a variable duration. As we mentioned above, we believe that the key questions to ask in characterizing how the learner moves through these stages are questions about his architecture for production. In particular we must ask what is the form of the learner's knowledge (e.g. *operator* vs. *search*

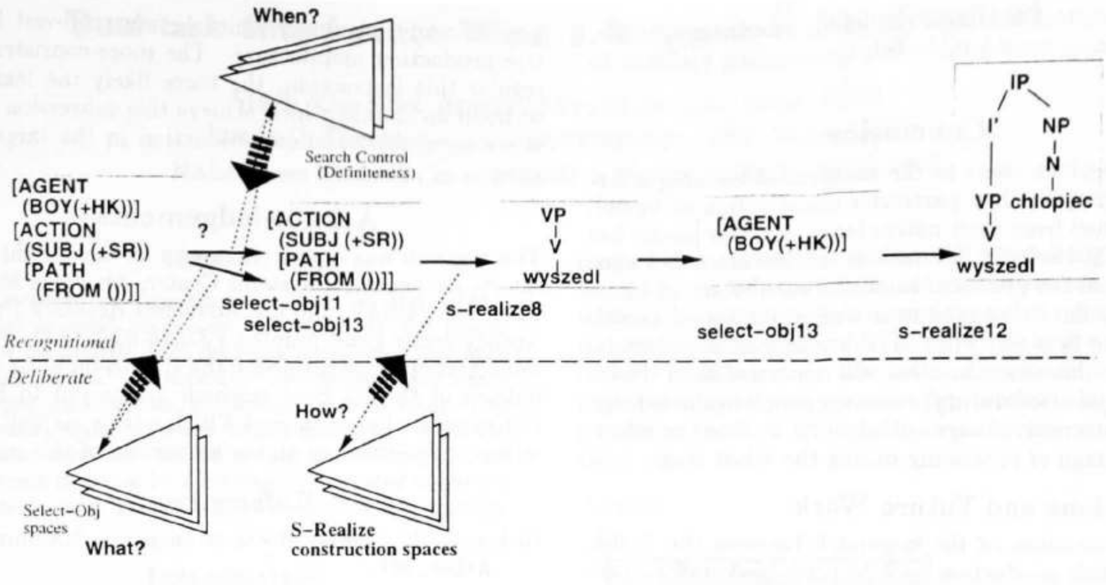

Figure 4: Polish generation of *wyszedl chlopiec*

control) and when is he able to bring that knowledge to bear during processing?

Omissions

The special problem that Polish learners of English face is that the organization of their production system requires them to apply knowledge of [HK,SR] at a time different from that required by the target language. In his native language, the Polish speaker uses knowledge of [+/-HK] to choose among alternatives, but this knowledge is not used in selecting which concepts to realize. He therefore comes to the task of English production with an architecture that is not designed to produce articles as separate lexical items.

This contrasts with the architecture for English production, wherein a speaker uses [+/-HK] in association with [+/-SR] when deciding which concepts to realize. For an English native speaker, the [HK,SR] knowledge becomes part of a chunked Select-Obj operator and as such, there is no opportunity for the lexical realization of [HK,SR] to be omitted. The task for the Polish learner then is to achieve a re-organization of his processing architecture so that [+/-HK] is no longer search control knowledge applied after semantic concepts have already been selected for realization, but is now compiled into the Select-Obj operators themselves. The amount of time it takes the learner to achieve this re-organization will be the duration of the omission stage in their Interlanguage.

Overgeneralizations

Although the learner who overgeneralizes articles has clearly not acquired the target language form, the mere fact that he produces them implies that he has achieved the necessary reorganization of knowledge. As we have discussed previously, the core of this reorganization comes from applying [+/-HK] knowledge at the Select-Obj (**what?**) stage, rather than at the search control (**when?**) stage. We would then predict that this is the primary feature they use for generating articles, perhaps producing them in all [HK] contexts. Recall that Polish native speakers use knowledge of [+/-SR] during the S-Realize stage of processing. Thus the English rules require shifting the point of application of this knowledge to an earlier stage and integrating it with [+/-HK] knowledge. Until the learner can combine these two features into a single Select-Obj operator, we expect overgeneralizations to occur.

Nonstandard Interlanguage Rules Overgeneralizations are one type of Interlanguage rule, but we might also expect a stage where the learner is more variable in his article production. Our model predicts this behavior because the selection of which concepts to realize (i.e. Select-Obj operators) is independent of the decision about how to realize them (i.e. S-Realize operators). Once the learner has achieved the knowledge transformations necessary to propose a Select-Obj operator for [+/-HK,+/-SR], he must still do separate processing to build up the chunked S-Realize operator for that concept. During this processing, the learner makes choices about which articles to apply and must also take into consideration other relevant knowledge such as countability and number. Hence even after the learner has made the knowledge transformation required to produce the correct left-hand-side of an Interlanguage rule, he still has a considerable feature-mapping problem to solve.

Conclusions

Native speakers come to the problem of learning a foreign language with a particular organization of knowledge learned from their native language. Our model has shown that the principle task of the learner is not simply to learn the grammar rules and vocabulary of a new language, but to engage in a reorganization of knowledge about how and when to realize linguistic universals. Primarily this reorganization will mean a shift in the locus of application during processing, applying knowledge that for the native language might be used at the **when** or **how** stage of processing during the **what** stage.

Predictions and Future Work

Our examination of the mismatch between the Polish and English production architectures makes clear predictions about the stages in acquiring English articles:

1) Articles will be omitted until the learner shifts his application of [HK] knowledge from search control to operator knowledge;

2) Overgeneralization rules of the form

```
[HK] --> the
```

will occur because [HK] knowledge is shifted first in response to the need to produce articles;[6]

3) Non-standard or inconsistent rules as the learner tries to correctly integrate the [+/-HK] and [+/-SR] features during the construction of Select-Obj operators.

These stages closely resemble those seen in the longitudinal study of a Laotian speaker done by (Huebner, 1983) and we are currently analyzing data on Polish learners to confirm our predictions for Polish. We also would like to extend our architectural model to other languages without articles (e.g. Chinese and Japanese) in order to understand what kinds of architectural mismatches cause problems which are more or less easy to overcome.

Implications for TEFL

Our architectural appoach sees the task of mastering a foreign language as one of mastering the production architecture for that language. For languages whose architecture is unlike that of the speaker's native language, this means reorganizing knowledge so that it can be used at different times in production. The obvious question that remains for teaching is how to facilitate this reorganization. As we explained in our brief description of NL-Soar's learning mechanism, *chunks* are created in response to *impasses*. When the learner reaches an impasse, there are any number of paths he might follow to achieve a solution. Direct instruction can steer the learner towards the right path, helping him locate the precise conditions under which he must subvert his native production architecture. The more consistent and regular this instruction, the more likely the learner is to build up chunks which achieve this subversion at the times necessary to allow production in the target language.

[6] Although the model does not directly predict *the* as the overgeneralized form, we adopt this because of the high frequency of *the* and because of the various studies that support the early appearance of this rule in Interlanguage. (Huebner, 1983; Tarone and Parrish, 1988)

Acknowledgements

This research was supported in part by the Wright Laboratory, Aeronautical Systems Center, Air Force Materiel Command, USAF, and the Advanced Research Projects Agency under grant number F33615-93-1-1330. The authors gratefully acknowledge the system-building contributions of the NL-Soar research group and to Robert DeKeyser for linguistic and TEFL advice, as well as excellent suggestions on an earlier version of this paper.

References

Bickerton, D. (1981). *Roots of language.* Karoma, Ann Arbor, MI.

Chauldron, C. and Parker, K. (1990). Discourse markedness and structural markedness: the acquisition of English NPs. *Studies in Second Language Acquisition*, 12:43–64.

Duskova, L. (1969). On sources of errors in foreign language learning. *International Review of Applied Linguistics in Language Teaching*, 7:11–36.

Huebner, T. (1983). *A longitudinal analysis of the acquisition of English.* Karoma Publishers, Ann Arbor.

James, C. (1969). Deeper contrastive analysis. *International Review of Applied Linguistics in Language Teaching*, 7:83–95.

Larson-Freeman, D. (1975). The acquisition of grammatical morphemes by adult ESL students. *TESOL Quarterly*, 9:409–419.

Lewis, R. L. (1996). Interference in short-term memory: The magical number two (or three) in sentence processing. *Journal of Psycholinguistic Research*, 25:93–115.

Lewis, R. L. and Lehman, J. F. (1997). Sentence processing with limited resources: A Soar-based architectural theory. *Psychological Review.* submitted.

Ringbom, H. (1985). The influence of Swedish on the English of Finnish learners. In Ringbom, H., editor, *Foreign language learning and bilingualism*, pages 39–71. Abo Akademi, Abo.

Szwedek, A. (1976). *Word order, sentence stress, and reference in English and Polish.* Linguistic Research, Edmonton, Alberta.

Tarone, E. and Parrish, B. (1988). Task-related variation in Interlanguage: the case of articles. *Language Learning*, 38:21–44.

Young, R. (1996). Form-function relations in articles in English Interlanguage. In Bayley, R. and Preston, D., editors, *Second Language Acquisition and Linguistic Variation*, pages 135–175. John Benjamins, Amsterdam.