Planning Tutorial Text in a System for Teaching English as a Second Language to Deaf Learners

Lisa N. Michaud and Kathleen F. McCoy
{michaud,mccoy}@cis.udel.edu
Computer and Information Sciences Department
University of Delaware Newark, DE 19716

Abstract

In this paper we discuss an envisioned text planner for a computer-assisted instruction tool for deaf learners of English. We describe the problem of deaf literacy and overview our system, designed to act as a writing tutor that generates text to explain the errors found in a user's written compositions. These explanations are created according to a response anatomy that divides the planning task into a bottom-up phase for grouping and ordering explanations and a top-down phase for building a hierarchical text plan to create the explanatory text. This is then followed by a revision of the complete plan to place it in its greater context. At all levels of planning, the explanation is personalized to the individual learner's styles and knowledge in order to maximize the learning experience.

Introduction

The problem of deaf literacy has been well-documented and has far reaching effects on every aspect of deaf students' education. Although data on writing skills is difficult to obtain, the reading comprehension level of deaf students has been established to be considerably lower than that of their hearing counterparts, "...with about half of the population of deaf 18-year-olds reading at or below a fourth grade level and only about 10% reading above the eighth grade level..."(Strong 1988) Instruction intended to close that gap must take into account the unique characteristics of the learner population. In our own observations, the writing samples we have collected from deaf individuals display a markedly different set of errors than those committed by native English speakers. Tailoring English instruction to the needs of these learners is the focus of our project.

This paper overviews one part of a system under development whose long-term goal is to be a "writing tutor" for deaf people who use American Sign Language. ICICLE (Interactive Computer Identification and Correction of Language Errors) is designed to accept as input an essay written by the user and then to generate text for a tutorial session addressing the errors found in the essay. The discourse generated as part of this session is the product of a response generation module that plans text according to a four-part response anatomy and that addresses user errors within the context of the user's language proficiency, his or her learning styles as exhibited through a history of interaction with the system, and past explanations. The result is a tutorial dialogue that is not only tailored to the learner but that also takes advantage of the dialogue history to exploit opportunities to use comparison and contrast to encourage "meaningful learning," tying new information together with recently-discussed and established knowledge (Brown 1994).

In this paper, after discussing further the motivation of the system by describing the challenges facing the deaf learner of English, we sketch an overview of the system design and describe the goals of the tutorial component. We then propose a model for the text planner which will construct system responses to the user. We will finish by comparing this to related work and listing those aspects of the planner which require further development.

Deaf Literacy

For some people who are deaf, American Sign Language (ASL) is the first language they acquire.[1] While the establishment of ASL as a native language provides great benefit in the acquisition of a second language, broad differences between the two languages such as those between ASL and English also pose challenges for the learner trying to transfer language knowledge from one to the other. ASL is a visual-gestural language whose grammar is distinct and independent of the grammar of English or any other spoken language (Stokoe 1960; Baker & Padden 1978; Baker & Cokely 1980; Hoffmeister & Shettle 1983; Klima & Bellugi 1979; Belman, Poizner, & Bellugi 1983). The sign-order rules of ASL are not the same as the word-order rules of English, and ASL syntax includes systematic modulations to signs as well as non-manual behavior (e.g., posture and facial expression) that achieves a simultaneous mode of communication

[1] We recognize that many people who are deaf or hard of hearing use other communication systems; however, we have chosen the population of native (or near native) users of American Sign Language as a target learner group.

not possible with the completely sequential nature of written English (Baker & Cokely 1980; Liddell 1980; Padden 1981; Klima & Bellugi 1979; Kegl & Gee 1983; Ingram 1978; Baker 1980).

Another obstacle to the ASL user acquiring English is the unique processing strategies he brings to the language task (Anderson 1993). The cognitive elements used to store signs in short-term memory are distinctly different from those used with a spoken/written language. Also, hearers of spoken language buffer the speech in order to process it together in words and phrases, but the buffer for visually observed data has a much quicker decay time than that of auditory data, which leads to repetition and redundancy in signed languages that does not occur in the same manner elsewhere. Moreover, long, involved utterances in a manual language are parceled into small pieces that are recursively reinforced, referring back to previous details as each new item of information is added, another characteristic atypical of spoken language. The native user of a manual language, therefore, is equipped with skills and expectations which are different from those needed to master a spoken/written language.

Added to these difficulties is the fact that ASL has no accepted written form, eliminating the opportunity to establish literacy skills in a fluent native language and then to transfer those skills to the new language being learned, far easier than acquiring literacy and a foreign language simultaneously (Lessow-Hurley 1996). Lastly, perhaps the worst difficulty for the deaf learner is that he or she has little to no understandable input in the "target" language; with poor or no aural capabilities, deaf learners receive nearly all of their English input through written material, often academic texts aimed at the comprehension level of their hearing peers (Anderson 1993), while the consensus among most researchers in Second Language Acquisition (Krashen 1982; Tarone 1982; Vygotsky 1986; Hatch 1983) holds that second language input at or near the learner's level of existing proficiency is most beneficial for learning.

In explaining the difficulties faced by the deaf learner of English, we do not propose that ASL natives are fundamentally different from other learners of English as a Second Language; rather, we want to stress the view that English is, for ASL natives, a distinctly different and challenging language, motivating the need to adopt a Second Language Acquisition strategy toward facilitating the learning process. There exist many obstacles to this process, some which are shared with other native language populations and some which are unique, such as the absence of the opportunity to have English input simplified to the personal level of acquisition and understanding of the learner. The system we propose attempts to address these needs as closely as possible within its own constraints (i.e., without the ability to converse with the learner in his or her native language).[2]

We should note that while there are "style checkers" and "grammar checkers" on the market, these programs do not satisfy the needs of the deaf. Educators of the deaf (and other people working with deaf individuals) report that such checkers frustrate deaf students. Geared for the writing style of fluent, native English speakers, they do not catch many errors that are common in the writing of people who are deaf, and, at the same time, they mistakenly flag many constructions that are not errors.

There have been some attempts to develop "grammar checker" systems specialized for deaf users. Perhaps the most notable of these is the system named Ms. Pluralbelle, which was developed and tested with students at Gallaudet University (Loritz 1990; Loritz, Parhizgar, & Zambrano 1993). The work described here differs from this earlier work mainly in its emphasis on correction and on its model of the user's acquisition process. More than a grammar checker, this system is a complex writing tutor with an intensely personalized approach. This is described further in following sections.

System Overview

Figure 1 contains a block diagram of the system under development. ICICLE is equipped with a user model for tracking the user's level of language acquisition, represented by the language features the user has learned and those he is in the process of acquiring. Incorporated in this model is also a history module for tracking how the varied tutorial techniques at its disposal succeed or fail with this user. Because of the depth of information built around each individual learner, the system is designed to be used over an extended period of time with several different pieces of writing. See (McCoy & Masterman (Michaud) 1997) for a more in-depth discussion of the entire system than is provided in this paper.

The system-user interaction in ICICLE centers around a cycle of user input and system response. The entry into the cycle occurs when the user inputs a piece of writing (either directly typed into the system, or through loading a prepared text file) for the system to analyze. With the help of the user model, the system identifies the errors in the text (McCoy, Pennington, & Suri 1996) and then trims the list of errors down to those relevant for tutoring. The determination of relevance relies on the model's representation of those language features which are above, below, and at the user's current understanding. Errors on language features that are well-known by the user are likely to be careless mistakes that do not require instruction, and errors on features far above the user's current proficiency level are not likely to be understood well enough for instruction to be worthwhile.

[2] Research into a model using signed on-screen output is currently underway, and this may result in a bimodal system in the future which is capable of giving some instruction directly in a signed language.

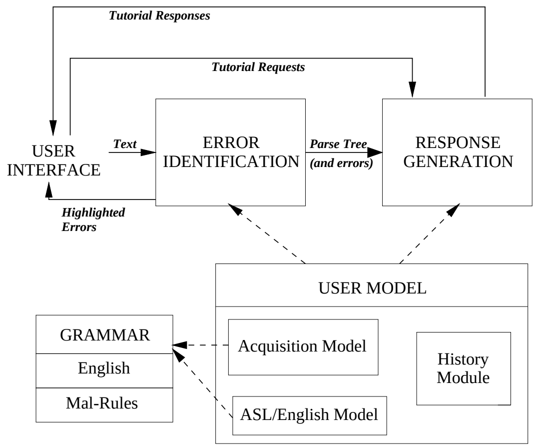

Figure 1: ICICLE System Architecture.

Once relevant errors have been identified, they are passed to a response generation module which brings the student through a review of those aspects of his or her writing which need to be improved, after which the student is free (and encouraged) to make changes and request a new analysis. This type of tutorial instruction, where the system reviews the performance of a user after a task has been completed, is motivated by the theory that the cognitive demands of some tasks are so intense that learning is hampered during their execution (Owen & Sweller 1985; Sweller 1988). It is our belief that the composition of original text in a non-native language is a task of this level of cognitive difficulty. Sources such as (Collins & Brown 1988) emphasize the usefulness of post-performance review or reflection as an approach to learning, and stress the suitability of computer systems in this kind of task, since they can perfectly capture the user performance and then review any aspect of it. Our system therefore endeavors to accomplish this goal and to optimize the knowledge derived from the composition experience by reviewing it with the user, an approach also found in some other language tutors such as MILT (Kaplan & Holland 1995).

While providing this tutorial feedback, ICICLE endeavors to satisfy the deaf learner's need for understandable input as mentioned above. We intend for our system's generated text to use grammatical constructions that involve those aspects of English the student is currently attempting to master. Another way in which we anticipate that our system will address the unique needs of the deaf population is by providing the user with feedback on his or her writing without involving a human teacher. Some students might prefer this mode of feedback since they would not risk feeling a "loss of face" as they might with a human tutor. The hope is that this will get the students to write more.

The current implementation of ICICLE captures the basics of the error identification process. The user interacts with the system through a window-based interface implemented in Tcl/TK. Analysis of the student's writing is performed by a bottom-up chart parser which is a successor to that in (Allen 1995). It makes use of an English grammar augmented with mal-rules (Sleeman 1982; Weischedel, Voge, & James 1978) derived from an error taxonomy compiled out of actual writing samples from deaf students (Suri 1993; Suri & McCoy 1993). These additional rules enable the grammar to recognize the errors expected from this target population. Errors found in the text are reported to the user by highlighting the original text with different colors corresponding to different types of errors. For these highlighted sentences, the user can elicit more information, including more specific highlighting of the problem area and a canned explanation of the type of error or errors. Users can fix the sentences accordingly and request re-analysis, but no user model or extended tutoring on the nature of the errors has yet been implemented.

This limited implementation allows for only a totally self-directed "learning environment," since the user (and not the system) selects which errors to explore. This may be quite useful for an advanced and motivated learner, but it is lacking in several respects. It makes no effort to build instruction around an overall plan to facilitate learning (see (Oxford 1995; Levy 1997; Kaplan & Holland 1995)). Also, the canned explanations leave no recourse if they are ineffective or if the student is not sufficiently motivated or knowledgeable to make the correction (see (Loritz 1995)). Finally, these limited explanations are entirely inflexible; they cannot be tailored to the user's level of expertise, they do not take into account the context of previous explanations that is so vital to human explanation, and they contain no comparison or contrast between similar errors. For these reasons, we are currently developing a more sophisticated model for system response.

Planning a Response

The purpose of ICICLE is to provide coherent explanations in the context of accessible and context-aware tutoring that is tailored to the individual. The text planner organizing these explanations must therefore operate on many levels to handle the various goals. A system response has multiple purposes: to address errors found in the user's writing in an order which maximizes absorption; to avoid addressing those errors which are beyond the user's understanding, or beneath his or her concern; to couch all instruction on the nature of the errors in a pedagogical approach that works well with this individual; to use language that is understandable; and to encourage meaningful learning through comparison and contrasting references that connect pieces of new and established information. We propose a text planning model that addresses each of these concerns.

This approach would make use of a multi-level planning element that divides its task into several phases. Given a group of tasks (errors to explain), it must execute a bottom-up strategy for grouping these tasks and

METHOD RULE: *State the grammar rule that was violated in the sentence.*

EFFECT:	(COMPETENT ?hearer (DO ?hearer (CORRECT ?hearer ?error)))
CONSTRAINTS:	(AND (BROKEN-RULE ?error ?rule) (GOOD-METHOD ?hearer TELL-RULE))
NUCLEUS:	(KNOW-ABOUT ?hearer (BROKEN-RULE ?error ?rule))
SATELLITES:	nil

Figure 2: Rough Draft of a Method Operator.

placing them in an order of execution. Then it must begin a top-down planning phase where each error to discuss represents a goal (to achieve a state where the user understands the error well enough to be able to correct it) and planning operators must be chosen to achieve that goal, each of which may create subgoals. The plan operators in this type of task represent the application of language toward achieving the communicative goal of the system, and will contain either rhetorical strategies which break down into subgoals or primitive speech acts. Planning is complete when all of the goals have been refined down to speech acts. The completely-specified text plan will be given to a sentence realization component so it can then be output as text.

This division of processing between determining an overall plan and performing lower-level discourse planning is consistent with that used in (Mooney 1993; Mooney, Carberry, & McCoy 1991; Woolf 1984) for planning very large discourse. Our approach is also similar to that of (Cawsey 1990; 1993) in that planning and execution (generation) may occur incrementally; some utterances will be generated and presented to the user before the plans for later errors are fleshed out. We have based the top-down, hierarchical aspect of our planner largely on the work described in (Moore & Paris 1992), modifying the planning operators outlined there for our purposes.

The Anatomy of a Response

The response generation model using this planning element is designed to generate text according to an "anatomy of a response" comprised of *content*, *method*, *form*, and *manner*. The term *content* refers to the error or errors being discussed in a given system action (explanation); the *method* is the choice between multiple possible pedagogical approaches to discussing the error; the *form* is the determination of the sentence-sized propositions (each containing a specific rhetorical force) that will eventually realize the method; and the *manner* refers to the discourse-level annotations that result in a cohesive, contextually-aware explanation. Once these phases are complete, the fleshed-out text plan will be sent to a sentence generation component to be realized as English. We discuss below how our proposed method incorporates each of these components in a separate "level" or component of planning.

Content

The *content* of an ICICLE explanation is the error being discussed; this is determined partly by the error identification module (see Figure 1), which selects only those errors relevant for discussion for delivery to the response generator.

The second part of the content determination is the grouping and ordering of the errors. Research in language pedagogy (Anderson 1993) and empirical studies on learning from written texts (Hayes-Roth & Thorndike 1979) both suggest that grouping together related information is more effective than explaining each error in the order in which it occurs in the essay. A domain knowledge base will contain information about the errors recognized by the system and possible grouping strategies to cluster them according to shared features. Next, the order of the clusters will be determined using information on how to best structure the overall discussion flow (Oxford 1995), completing the bottom-up phase.

Method

The next part of the planning process (and the first part of building the top-down plan) is selecting a tutorial *method* for addressing each error. Given the top-level goal of addressing a given error, the system must now begin building the top-down plan to accomplish that goal through text. We plan to implement several possible tutorial methods, based on research in second language pedagogy. Each of these approaches may appeal to a different style of language learner. Among the possibilities may be:

- To simply provide a corrected form of the sentence.
- To explain the grammar rule that was broken by this error. (A sample planning operator addressing this choice is shown in Figure 2.)
- To provide examples of sentences that illustrate correct usage of this grammar rule.
- To compare and contrast the grammar rule involved with its corresponding rule in ASL.

These methods would be represented in our planner by planning operators. We currently have some very rough drafts of potential operators that will be used to

```
FORM RULE:    A short statement of the grammar rule.

EFFECT:       (KNOW-ABOUT ?hearer (BROKEN-RULE ?error ?rule))
CONSTRAINTS:  (AND (FAMILIAR ?hearer (CONCEPT ?rule))
                   (NUCLEUS))
NUCLEUS:      (INFORM ?speaker ?hearer (BROKEN-RULE ?error ?rule))
SATELLITES:   nil
```

```
FORM RULE:    A Long statement of the grammar rule.

EFFECT:       (KNOW-ABOUT ?hearer (BROKEN-RULE ?error ?rule))
CONSTRAINTS:  (AND (UNFAMILIAR ?hearer (CONCEPT ?rule))
                   (COMPETENT ?hearer (DO ?hearer (LEARN ?hearer (CONCEPT ?rule))))
                   (NUCLEUS))
NUCLEUS:      (INFORM ?speaker ?hearer (BROKEN-RULE ?error ?rule))
SATELLITES:   (KNOW-DEF ?hearer (CONCEPT ?rule))
```

Figure 3: Rough Draft of Form Operators to State Rule.

illustrate the steps of the planning process. An example can be found in Figure 2. The fields of the operator are **EFFECT** (the goal the operator can be used to achieve), **CONSTRAINTS** (those things which must be true before the operator can be used), **NUCLEUS** (the main subgoal or action represented by this operator), and **SATELLITES** (one or more additional subgoals, some of which may be optional).

Given a goal, the planner will search for operators whose **EFFECT** fields match that goal. At the *method* level of planning, the system's goal is to make the hearer competent to correct the error. Thus it will be looking for operators with **EFFECT** fields like the one in Figure 2: `(COMPETENT ?hearer (DO ?hearer (CORRECT ?hearer ?error)))`.

The choice between these methods is motivated by the user model's knowledge information – what the user knows and what he or she is likely to understand – and history information – the user's long-term performance. The operator shown in Figure 2, would be chosen if explanations of the grammar rule (the TELL-RULE method) have been shown to be successful with this user.[3] Of particular interest is the success or failure of past methods chosen, as measured by subsequent performance on revisions and other pieces of writing. Over time, the system should be able to make reasonably principled decisions on what style of instruction is best suited to the individual.

Form

The selection of the higher-level planning operator representing the method posts the subgoals contained in the **NUCLEUS** and **SATELLITES** fields of that operator. The processing of these subgoals comprises the *form* phase of the planning model.

At this time, the chosen method may be executed in different ways based on the user's expertise. A sample operator for planning this level can be seen in top of Figure 3. If the user is familiar with the concept of the grammar rule involved, the system can just **INFORM** the user of the type of error. An example of this would be: *This sentence contains an error in subject-verb agreement.*

Alternatively, the user model may indicate that the user is *not* familiar with the concept of the grammar rule, so the constraints of that operator would not be satisfied. An alternative is shown in the bottom of Figure 3, which would be applicable if the user is unfamiliar with the rule but currently capable of learning explicitly about it. In this operator, the same nucleus exists for stating the grammar rule, but now a satellite posts the subgoal to define the rule. This will create the goal of explaining what subject-verb agreement is. Achieving this goal may in turn generate the definition of subconcepts involved in the explanation (depending upon the knowledge of the user). A sample text generated by this process could be: *This sentence contains an error in subject-verb agreement. In English, third-person singular subjects (pronouns like HE, SHE, and IT, singular noun phrases like THE DOG, and names like JOHN) require a present tense verb to have the agreement marker -S at the end.*

We note here that we do not expect our user model to always be infallible or complete. Because of this, we will be following in the footsteps of (Cawsey 1990; 1993) and (Carenini & Moore 1993), storing all decisions based on the user model in our plan as it is built. This information can be used later to recover from incorrect user model information should an explanation

[3]We expect our operators to have, in general, much more detailed constraint lists than are shown in these examples, referencing knowledge in the user model, history module, domain knowledge base, and dialogue history.

fail. Note, however, that in previous work, only assumptions that the user knows more than the model indicates are made and recorded; our user model, being very complex and dynamic, has the potential of either underestimating *or* overestimating the user. Because of this, it may be wise to record all decisions that are based on information in the user model *and* all decisions based on the possibility the model is wrong. When recovering from failed explanations, our user model may need to be updated to compensate. This is a matter for further exploration.

The text plan at the completion of this phase contains the basic linguistic structure of the discourse, molded by the tutorial method chosen and fleshed out with prerequisite data. The next step in the process is to place this discourse within its context.

Manner

The primary job of the *manner* component is to tie new knowledge into existing knowledge. In order to do this, the manner component must now create a context-aware explanation, a process we view as a revision of the existing plan. It has been observed (Moore 1993) that comparison and contrast to recent and established material is a powerful tool of human-generated explanations, and essential for generating comprehensible tutorial discourse. At this time, therefore, the text planner needs to begin to make discourse-level adjustments to its plan in order to insert comparison goals where warranted.

The context which needs to be exploited for this step is found in three tiers of proximity: the information discussed within this group of explanations, the information discussed earlier in a session, and established information the user already knew. These comparisons may have distinct sources of information for the references they represent, which would be reflected in the phrases generated from them. Recent explanations would be referred to in a phrase such as, "Like X discussed above..." An explanation from earlier in the session would be referred to in a phrase like, "You may remember when I talked about Y..." The student's existing knowledge can be referenced by saying, "This is like Z.". The constraints of the operator representing the rhetorical strategies that implement the reference, would pinpoint which source is appropriate for the phrase: explanations from the current response are in the text plan at hand; recent explanations from earlier in the current session can be accessed through the history module, in which the plans are stored to show the dialogue history with this student; and the student's existing knowledge is available in the user model. Identifying which pieces of previous discourse to use in references and when references should be inserted is another matter for further exploration.

Generating from the Completed Plan

As mentioned above, ICICLE is envisioned as a system that would very closely approximate the language level of its user in all generated text; as such, it needs to be able to generate the same rhetorical idea using a variety of levels of syntactic complexity. Therefore, our model cannot leave the choice of syntactic structures to chance or to indirect constraints in a text generation component. It is our belief that FUF, a functional unification-based system (Elhadad 1993), can be modified for work with our approach. Alternative possible syntactic structures for a given sentence would be ordered according to the acquisition model for this user. Those constructions that are at the user's learning level will be preferred by ordering them before other potential realizations. If the user model is updated and the acquisition model changes, a new ordering of the alternatives would reflect this change. The ordering would affect which syntactic structure is chosen at this phase of the process. In this way, the system will help provide the understandable input so needed by this learner population.

Related Work

We have stressed before the inspiration we received from the work of Moore and Paris on the Explainable Expert System, but our work is distinct from theirs in several ways, most noticeably the complexity of user modeling. In work describing the systems in which the EES has been implemented, no mention is made of updating the user model after the initial establishing effort, whereas our user model needs to capture the progression of knowledge and language capabilities. We also track our users across multiple experiences with the system, leading to the need for a history module to hold information about explanation efforts beyond the current session.

Other related work that has influenced ICICLE design choices includes the use of referring expressions to establish context described in the work on the Sherlock (Moore 1993; Rosenblum & Moore 1993) and Migraine (Carenini & Moore 1993) systems. We are greatly attracted by the idea of linking current explanations to prior explanations. In that work, however, the implementation of contextual references was done explicitly in the planning operators that build the original text plan; the creators admit that this leads to a preponderance of planning operators to cover all of the communicative goals paired with all of the possible backward references. We prefer to view these references as annotations or improvements upon a basic plan; therefore, it makes sense to do them in two separate phases, first sketching out the basic plan, and then fleshing it out with appropriate comparison references.

An implemented system which seems on the surface to have a lot in common with ICICLE is Reva Freedman's CIRCSIM (Freedman 1996b; 1996a), an intelligent tutoring system for first year medical students.

Like ICICLE, CIRCSIM reviews a task with a student after it has been finished with the aim of correcting errors detected by the system. Freedman's system also implements multiple pedagogical approaches and varied syntactic realizations; however, the multiple choices in CIRCSIM's text planner are implemented solely for the sake of creating variation in the text, and are not motivated by any goal of meeting the needs of an individual student. In fact, there is no agenda in CIRCSIM to tailor its explanations to the user; there is no user model or history model. Because of the learner population for ICICLE and its unique needs, our planning approach is necessarily more complex.

Conclusion and Future Work

In this paper, we have discussed our text planner model for our proposed tutoring system. We explained our multi-level planning design, including a four-part "response anatomy" that composes tutorial text within the context of previous explanations and the user's knowledge, learning style, and history with the system. We explained how we hope this system will address some of the needs of the deaf learner population.

Future research will involve the completion of our planning operator design, work on identifying relevance in prior explanations for generating comparison text, and completion of our presentation model for how these responses will be given to the user and how the user's reaction in terms of acceptance or exhibiting a failure to understand will be handled.

Acknowledgments

This work has been supported by NSF Grant #IRI-9416916, NSF Research Traineeship Grant #GER-9354869, and a Rehabilitation Engineering Research Center Grant from the National Institute on Disability and Rehabilitation Research of the U.S. Department of Education (#H133E30010).

References

Allen, J. 1995. *Natural Language Understanding*. California: Benjamin/Cummings, 2nd edition.

Anderson, J. J. 1993. *Deaf Student Mis-Writing, Teacher Mis-Reading: English Education and the Deaf College Student*. Burtonsville, MD: Linstok Press.

Baker, C., and Cokely, D. 1980. *American Sign Language: A Teacher's Resource Text on Grammar and Culture*. Silver Spring, MD: TJ Publishers.

Baker, C., and Padden, C. 1978. Focusing on the non-manual components of American Sign Language. In Siple, P., ed., *Understanding Language through Sign Language Research*. New York: AP. 27–58.

Baker, C. 1980. Sentences in American Sign Language. In Baker, C., and Battison, R., eds., *Sign Language and the Deaf Community*. Silver Spring, MD: National Association of the Deaf. 75–86.

Belman, K.; Poizner, H.; and Bellugi, U. 1983. Invariant characteristics of some morphological processes in American Sign Language. *Discourse Processes* 6:199–223.

Brown, H. D. 1994. *Principles of Language Learning and Teaching*. Englewook Cliffs, NJ: Prentice Hall Regents, 3rd edition.

Carenini, G., and Moore, J. D. 1993. Generating explanations in context. In Gray, W. D.; Hefley, W. E.; and Murray, D., eds., *Proceedings of the International Workshop on Intelligent User Interfaces*, 175–182. Orlando, Florida: ACM Press.

Cawsey, A. 1990. Generating explanatory discourse. In Dale, R.; Mellish, C.; and Zock, M., eds., *Current Research in Natural Language Generation*. London: Academic Press.

Cawsey, A. 1993. *Explanation and Interaction: The Computer Generation of Explanatory Dialogues*. Cambridge, MA: MIT Press.

Collins, A., and Brown, J. S. 1988. The computer as a tool for learning through reflection. In Mandl, H., et al., eds., *Learning Issues for Intelligent Tutoring Systems*. NY: Springer-Verlag. chapter 1, 1–18.

Elhadad, M. 1993. *FUF: The Universal Unifier User Manual Version 5.2*. Columbia University, Computer Science Department.

Freedman, R. 1996a. Using a text planner to model the behavior of human tutors in an its. In *Online Proceedings of the 1996 Midwest AI and Cognitive Science Conference*. URL www.cs.indiana.edu/event/maics96/Proceedings/Freedman/freedman.html.

Freedman, R. 1996b. Using tutoring patterns to generate more cohesive text in an intelligent tutoring system. In Edelson, D. C., and Domeshek, E. A., eds., *Proceedings of the International Conference on the Learning Sciences (ICLS '96)*, 75–82. Evanston, IL: Association for the Advancement of Computing in Education.

Hatch, E. 1983. Simplified input and second language acquisition. In Andersen, R., ed., *Pidginization and Creolization as Language Acquisition*. Newbury House. 64–86.

Hayes-Roth, B., and Thorndike, P. W. 1979. Integration of knowledge from text. *Journal of Verbal Learning and Verbal Behavior* 18:91–108.

Hoffmeister, R. J., and Shettle, C. 1983. Adaptations in communication made by deaf signers to different audience types. *discourse processes* 6:259–274.

Ingram, R. M. 1978. Theme, rheme, topic and comment in the syntax of American Sign Language. *Sign Language Studies* 20:193–218.

Kaplan, J. D., and Holland, V. M. 1995. Application of learning principles to the design of a second language tutor. In Holland, V. M.; Kaplan, J. D.; and Sams, M. R., eds., *Intelligent Language Tutors:*

Theory Shaping Technology. Mahwah, New Jersey: Lawrence Erlbaum Associates. 273–287.

Kegl, J., and Gee, P. 1983. Narrative/story structure, pausing and American Sign Language. *Discourse Processes* 6:243–258.

Klima, E. S., and Bellugi, U. 1979. *The Signs of Language*. Cambridge, MA: Harvard University Press.

Krashen, S. 1982. *Principles and Practice in Second Language Acquisition*. Oxford: Pergamon Press.

Lessow-Hurley, J. 1996. *The Foundations of Dual Language Instruction*. New York: Longman Publishers USA, 2nd edition.

Levy, M. 1997. *Computer-Assisted Learning Context and Conceptualization*. New York: Oxford University Press.

Liddell, S. K. 1980. *American Sign Language Syntax*. The Netherlands: Mouton Publishers.

Loritz, D.; Parhizgar, A.; and Zambrano, R. 1993. Diagnostic parsing in call. *CAELL Journal* 1(4):9–12.

Loritz, D. 1990. Using artificial intelligence to teach english to deaf people. Technical Report RX2500-950F, Language Research Laboratories, Georgetown University.

Loritz, D. 1995. Gpars: A suite of grammar assessment systems. In Holland, V. M.; Kaplan, J. D.; and Sams, M. R., eds., *Intelligent Language Tutors: Theory Shaping Technology*. Mahwah, New Jersey: Lawrence Erlbaum Associates. 121–133.

McCoy, K. F., and Masterman (Michaud), L. N. 1997. A tutor for teaching english as a second language for deaf users of american sign language. In *Proceedings of Natural Language Processing for Communication Aids, an ACL/EACL97 Workshop*, 160–164.

McCoy, K. F.; Pennington, C. A.; and Suri, L. Z. 1996. English error correction: A syntactic user model based on principled mal-rule scoring. In *Proceedings of User Modeling '96*.

Mooney, D.; Carberry, S.; and McCoy, K. 1991. Capturing high-level structure of naturally-occurring, extended explanations using bottom-up strategies. *Computational Intelligence: Special Issue on Natural Language Generation* 7(4).

Mooney, D. J. 1993. *Generating High-Level Structure for Extended Explanations*. Ph.D. Dissertation, University of Delaware.

Moore, J. D., and Paris, C. L. 1992. Planning text for advisory dialogues: Capturing intentional and rhetorical information. *Computational Linguistics* 19(4):651–695.

Moore, J. D. 1993. What makes human explanations so effective? In *Proceedings of the 15th Annual Meeting of the Cognitive Science Society*. Hillsdale, NJ: Lawrence Erlbaum Associates.

Owen, E., and Sweller, J. 1985. What do students learn while solving mathematics problems? *Journal of Educational Psychology* 77:272–284.

Oxford, R. L. 1995. Linking theories of learning with intelligent computer-assisted language learning (icall). In Holland, V. M.; Kaplan, J. D.; and Sams, M. R., eds., *Intelligent Language Tutors: Theory Shaping Technology*. MahWah, New Jersey: Lawrence Erlbaum Associates. 359–369.

Padden, C. 1981. Some arguments for syntactic patterning in American Sign Language. *Sign Language Studies* 32:239–259.

Rosenblum, J. A., and Moore, J. D. 1993. Participating in instructional dialogues: Finding and exploiting relevant prior explanations. In *Proceedings of the World Conference on Artificial Intelligence in Education*.

Sleeman, D. 1982. Inferring (mal) rules from pupil's protocols. In *Proceedings of ECAI-82*, 160–164. Orsay, France: ECAI-82.

Stokoe, W. C. J. 1960. Sign language structure. *Studies in Linguistics occasional papers* 8.

Strong, M. 1988. A bilingual approach to the education of young deaf children: ASL and English. In Strong, M., ed., *Language Learning and deafness*. Cambridge: Cambridge University Press. 113–129.

Suri, L. Z., and McCoy, K. F. 1993. A methodology for developing an error taxonomy for a computer assisted language learning tool for second language learners. Technical Report TR-93-16, Dept. of CIS, University of Delaware.

Suri, L. Z. 1993. *Extending Focusing Frameworks to Process Complex Sentences and to Correct the Written English of Proficient Signers of American Sign Language*. Ph.D. Dissertation, University of Delaware. Available as Dept. of CIS Technicial Report TR-94-21.

Sweller, J. 1988. Cognitive load during problem solving: Effects on learning. *Cognitive Science* 12:257–285.

Tarone, E. 1982. Systematicity and attention in interlanguage. *Language Learning* 32(1):69–84.

Vygotsky, L. S. 1986. *Thought and Language*. Cambridge, MA: MIT Press.

Weischedel, R. M.; Voge, W. M.; and James, M. 1978. An artificial intelligence approach to language instruction. *Artificial Intelligence* 10:225–240.

Woolf, B. P. 1984. Context dependent planning in a machine tutor. Technical Report COINS Technical Report 84-21, Dept. of Computer and Information Science, University of Massachusetts.