SESSION VII

CODING AND ANALYSIS OF DATA

Ron D. Hays, *President*
RAND Corporation

The CHIP framework: Automatic coding and analysis of parent-child conversational interaction

JEFFREY L. SOKOLOV and BRIAN MACWHINNEY
Carnegie-Mellon University, Pittsburgh, Pennsylvania

CHIP is a computer program for the automatic coding and analysis of parent-child conversational interaction. The program was developed because manual coding of large collections of computerized transcripts cannot possibly be completed within a reasonable time span. CHIP codes parent-child conversational data as stored in transcript files and computes a series of descriptive statistics based on these codes. Parental responses to child utterances and child responses to parent utterances are both coded. This allows for an analysis of the reciprocal relationship between parental and child language. Three longitudinal corpora from CHILDES (totaling 151,900 utterances) were coded and analyzed by CHIP. The results indicated a high degree of contingency between parental and child language for different word classes across a large span of development. Two main points are argued: (1) Automatic data coding and analysis programs are important new tools for transcript analysis, and (2) CHIP, as an example of such a tool, can provide detailed information concerning the exact nature of parent-child conversational interactions.

Our goal in this paper is to present computational tools for exploring the role of parent-child conversational interaction in language development. After presenting a brief history of the theoretical controversy surrounding this issue, we will examine the potential role of conversational interaction in language learning. We will argue that certain sources of conversational input provide children with excellent opportunities to advance their linguistic skills. These sources will be characterized in terms of a typology of interactions between parents and their children that may be coded and analyzed automatically by using a program called CHIP. Following a description of CHIP, we will present some analyses of parent-child conversational interaction.

A Brief Historical Overview

The most recent revival of the nature/nurture debate can be traced back to Skinner's (1957) claim that children learn language through contingent feedback. According to Skinner, language is taught to children according

to the principles of operant conditioning. He argued that children are positively reinforced for correct language usage and negatively reinforced for incorrect usage; in this manner, children come to produce only grammatical speech. Chomsky (1959) showed the theoretical problems with this position. Even more importantly, Brown and Hanlon (1970) showed it to be empirically false. They looked through transcripts of mother-child interaction to see if mothers provided explicit feedback to their children regarding the grammaticality of their speech. Candidates for positive feedback were statements such as *Well done* or *That's right* following grammatical utterances and *No* or *That's not right* following ungrammatical ones. They discovered that parents did not provide children with explicit feedback that was contingent upon the grammaticality of their speech. Instead, the explicit feedback that parents did provide to their children was contingent on the truth value of the child's speech. For example, when the child Adam produced *Draw a boot paper*, his mother positively reinforced the truth of the utterance by responding *That's right. Draw a boot on the paper* while seemingly ignoring its ungrammaticality.

At about the same time, Gold (1967) demonstrated mathematically that only finite state or regular languages are learnable from positive data alone. Gold's proof high-

Correspondence may be addressed to Jeff Sokolov, Graduate School of Education, Larsen Hall, Third Floor, Harvard University, Cambridge, MA 02138.

lights the learnability tradeoffs that exist between the form of input that children may receive and the constraints that may exist on types of allowable grammars. Simply stated, even if children do not receive negative data, learnability may be demonstrated by constraining the range of grammars that they may hypothesize during learning. Thus the two major findings for learnability were that children do not receive explicit negative feedback from their parents, and that non-finite-state languages are not learnable without negative feedback. These two findings provided indirect support for the nativist position. The nativists could argue that since natural languages are not merely finite-state automata, and since the input that children receive is not sufficient for language to be learnable, certain aspects of language must be innate. Following these results, several nativist theories of the acquisition of language were proposed. Each and every theory is based on the claim that children do not receive negative feedback. This is currently being called "the no-negative-evidence hypothesis" by Roeper and Williams (1987) and others.

Interactional Sources for Learning

As stated in the previous section, a major cornerstone of the argument in favor of specifically linguistic, innate constraints lies in the supposed lack of evidence for negative feedback. However, several recent studies have discovered that parents do respond differentially to the grammatical and ungrammatical utterances produced by their children. These differential responses can be enough to signal an error to the child. For example, Hirsh-Pasek, Treiman, and Schneiderman (1984) have found that parents tend to repeat more of their children's ill-formed utterances than well-formed ones. They have also discovered that these repetitions often include corrections. Demetras, Post, and Snow (1986) have discovered that the most common response to a well-formed child utterance is to continue on with the conversational topic, whereas the most common response to an ill-formed utterance is a clarifying question. In addition, Warren-Leubecker, Bohannon, Stanowicz, and Ness (1986) have noted that parents use repetition to respond to syntactic (29% of the time) and phonological (21% of the time) errors. Furthermore, this effect is modified by the length of the child's utterance. There is more of a tendency for parents to repeat longer utterances than shorter ones. In addition, repetitions tend to occur more often when only one child error is produced. This indicates that parents most often provide children with differential feedback when only one error is produced and when their utterance is at the upper boundary of their abilities. Finally, numerous intervention studies by Keith Nelson and his colleagues (Baker & Nelson, 1984; Nelson, 1977; Nelson, Carskaddon, & Bonvillian, 1973; Nelson, Denninger, Bonvillian, Kaplan, & Baker, 1984) suggest that these expansions or recasts can and do facilitate language learning.

In fact, even in the examples that Brown and Hanlon (1970) used to illustrate the actual uses of explicit approval and disapproval by parents (see their Table 1.12, p. 49), it is possible to find examples of recasts providing im-

plicit negative feedback. Reconsider the above example, in which Adam says *Draw a boot paper*, and his mother responds *That's right*; in the sequence of utterances surrounding this interaction within the transcript, there are repeated examples of Adam's mother recasting his incorrect utterance into *Draw a boot on the paper*.

The basic point is that children may receive many different forms of facilitative input from their parents. Children receive implicit negative feedback concerning their errors:

- *SAR: He no bit ya.
- *MOT: No # he won't bite ya. (1)

Or parental expansions may provide children with positive examples of correct usage:

- *MOT: What happened to your shirt?
- *ADA: Got some choc(olate).
- *MOT: You got some chocolate on it # didn't you? (2)

Or they may reveal the membership of various lexical items in certain grammatical categories:

- *ABE: What you got in your hand?
- *FAT: What have I got in my hand? (3)

In Example 3, the parental model reveals to the child that the pairs of *you-I* and *your-my* are grammatically interchangeable. In addition, parental partial repetition of lexical elements from child speech may serve as communicative and cognitive facilitators. Partial repetition may be simultaneously a communicative facilitator by enhancing topic maintenance and a cognitive facilitator by reducing memory demands (i.e., lexical access), thereby freeing up more resources for the child's learning.

It would be interesting to examine parental additions to adjacent child speech, in order to determine the rate and pattern of examples provided to the child by the parent, and to compare these data with the corresponding values for child deletions of adjacent parental speech. This would provide information about how the amount and pattern of parental input might change as a function of child development. Evidence for this form of contingent input would provide further support for the importance of input in language development. In order to perform this analysis on large corpora of data (50,000 utterances), an automatic data coding and analysis program was designed and implemented.

THE CHIP FRAMEWORK

CHIP is a computer program for the automatic coding and analysis of parent-child conversational interactions. The program is intended to utilize the data and tools of the Child Language Data Exchange System (CHILDES) as described in MacWhinney (in press). CHIP assumes that data is transcribed according to the CHAT conventions also described by MacWhinney. The program was developed to be consistent with the design guidelines

(down to the actual code) of the CLAN data analysis programs developed for and distributed through CHILDES.

The establishment of CHILDES (MacWhinney & Snow, 1985) has greatly facilitated the creation of an automatic data analysis system. Researchers now have available to them a tremendous amount of data for analysis. Automatic data analysis makes it possible for child language researchers to utilize the power in the CHILDES database more fully. There are several obvious advantages and disadvantages to automatic data coding relative to manual coding. The four major advantages are: (1) The coding criteria are explicit, because they are codified in the written program; (2) the coding process is reliable and not subject to variances across time and coders; (3) larger speech samples can be analyzed more quickly; (4) the entire process can be replicated by other investigators. The disadvantages are: (1) It takes great time and effort to design and implement an automatic coding system; (2) manual coding is naturally (rather than artificially) intelligent and can be more sensitive to subtle coding distinctions. Manual and automatic coding are not necessarily independent processes. It is possible, for example, to use automatic coding to locate particular observations for more refined manual coding later on.

CHIP takes as input a computer file containing conversational interaction and adds to it a series of codes describing the similarities and differences between pairs of utterances spoken by different interlocutors. These utterances are called the *source* and the *response* utterances. The description of the program in the following sections is divided into three major categories: (1) the coding system; (2) the structure of the program; and (3) the automatic coding and analysis process.

The CHIP Coding System

The CHIP program looks at pairs of utterances. The first utterance is the source utterance and the second utterance is the response utterance. CHIP compares the response utterance to the source utterance and inserts special codes into the child language transcript as a distinct coding tier. In addition, the program tabulates a series of descriptive statistics as it processes a data set. These will be described in a later section.

The following example should provide the reader with a gentle introduction to the coding system. Given the following mother-child interchange:

*CHI: I'm sitting down.

*MOT: You're sitting in your chair.

the CHIP system would produce the following tiers:

*CHI: I'm sitting down.

*MOT: You're sitting in your chair.

%adu: \$EXA:sitting \$ADD:in-your-chair
\$SUB:you're \$DEL:down
\$REP = 0.40

The coding tier indicates that the adult response contained an EXAct-match of *sitting*, a continuous ADDition

of *in-your-chair*, a DELetion of *down*, and a SUBStitution of the subjective pronoun *you're* for *I'm*. Finally, a repetition index is computed for the amount of overlap between the source and response utterances. The repetition index (counting the substitution) is 0.40 (2 overlapping words divided by 5 total words in the response). Note that the program is able to code pronoun substitutions only when a list of pronouns is provided. The idea is that each separate list of lexical items is meant to signify a lexical class for which substitution may be legal in the service of some linguistic function.

Three coding tiers are possible:

%adu: The coding tier containing child-adult source-response pairs.

%chi: The coding tier containing adult-child source-response pairs.

%slf: The coding tier containing child-child or adult-adult source-response pairs.

The complete coding system is as follows:

\$ADD: Additions of *N* continuous words.

\$DEL: Deletions of *N* continuous words.

\$EXA: Exact-matches of *N* continuous words.

\$SUB: Substitutions of *N* continuous words from within a specified word list.

\$NO__REP: The source and response do not overlap.

\$LO__REP: The overlap between source and response is below a user-specified minimum.

\$EXACT: Source-response pairs with no additions or deletions.

\$EXPAN: Source-response pairs with exact matches and additions but no deletions.

\$REDUC: Source-response pairs with exact matches and deletions but no additions.

\$SUBST: Source-response pairs with only exact matches and substitutions.

\$REP: The percentage of repetition between source and response ($0.0 \leq N \leq 1.0$).

\$FRO: The word-class item has been fronted.

The Structure of the Program

The program has four modules: a toolkit of utilities from CLAN, a series of data structures and operations for analyzing words, a series of data structures and operations for analyzing utterances, and a series of data structures and operations for analyzing source-response utterance pairs. In computer science, a data structure is simply a structure defined so that different levels of data about a single object can be stored and accessed meaningfully. The data structure for words includes fields for the word itself and its components (stem and affixes). The data structure for utterances includes fields for the identity of the speaker, the speaker's class (adult or child), and the words in the actual utterance. The data structures for the interactional data contain fields for all the summary statistics traced by the program. When CHIP is invoked, it is told by a series of flags who the child and adult participants are and what options to code. The CLAN utilities are

subroutines for handling the coding conventions provided by CHAT. These utilities distinguish different aspects of the coding system, such as speaker identifications, text tiers, and coding tiers. Without these coding conventions, child and adult utterances could not be distinguished.

The word level. The most important question to ask at this level is "What counts as a word?" Any string of characters delineated by spaces or the standard punctuation markers counts as a word, with the exception of the following: xx, xxx, yy, yyy, www, zzz, strings prefixed by 0 or &, and any retraces without corrections (e.g., "The <elephant is coming> [/] elephant is coming"). Everything else counts as a word; no attempt is currently made to delete other non-meaning-bearing strings. This definition is basically the one provided by CLAN.

The utterance level. The most important question to ask at this level is, "What counts as an utterance?" The CHIP program assumes that each main text tier has only one utterance on it. To ensure this, the CLAN utility *fixit* is used to separate all multiple utterance tiers into individual utterances. According to *fixit*, utterances are delimited by the following symbols: . ! ? +... and +\ . This is pre-defined by CHAT convention (for further detail, consult the CHAT manual). Main text tiers with no text are correctly ignored. Otherwise, the utterance level has two major functions: to classify the speaker (adult or child), and to store the utterance for the interactional analysis to come later.

The interactional level. At this level, data consists of a pair of utterances with a specific relation to one another: adult-child, child-adult, child-child, or adult-adult.

No internal language-specific knowledge. There are also several forms of knowledge one might provide to a program like CHIP. These might range in difficulty from building a front end with a full-blown part-of-speech tagger to simply maintaining a list of affective interactional markers (e.g., *huhuh* or *yeah*). Somewhere in between, one could find modules for parsing English morphology or for expanding contractions. The problem with morphology and contractions is, of course, ambiguity. CHAT eliminates this problem, however, by providing codes for transcribing morphemes and contracted elements in disambiguated form. So building these difficult modules within the CHIP framework would not be an efficient use of resources. Even so, CHIP does not currently utilize these coding conventions. The reason for this is practical: Many of the corpora analyzed for this article were not coded in full adherence to current CHAT convention and do not disambiguate morphology and contractions, although they are under development. The program does make one attempt at allowing users to use language-specific knowledge. By maintaining the option of including a list of words for a focused analysis, CHIP provides for the possibility of analyses of word-class differences or even categorization processes. The matching procedure could be characterized as a word-based rather than a code-based system. As such, CHIP takes its input from text that is based on spoken words rather than on coded symbols.

The Automated Coding Process

CHIP moves through transcripts of conversational interaction looking for appropriate source-response utterance pairs to code (child-adult, adult-child, child-child, or adult-adult). The source utterance is the reference utterance to which any responses are made. When a response utterance is encountered, the program looks backward (through a window of seven utterances) for the most recent and only the most recent potential source utterance. Only one source utterance is coded for each response utterance. If a source utterance cannot be found within the seven-utterance window, the program simply moves forward in the file, looking for a different response. Once a source-response pair has been found, a simple matching procedure is performed.

The matching procedure first looks for exact matches. Then, if substitutions are being coded (i.e., both the substitution flag and an include list have been provided), any words in the response that have a match in the include list are coded as substitutions. Following this, all non-overlapping words in the response are coded as additions, and all nonoverlapping words in the source are coded as deletions.

After the matching has been performed, if the amount of overlap is zero or below the minimum specified by the user (default is zero), then no statistical information is computed and either the code \$NO__REP or the code \$LO__REP is inserted into the coding tier. It is important to note here that a substitution is counted as an overlap. Although this may seem strange for the open-class lexical items, it is important for the closed-class ones because of possible shared meanings or functions between lexical items. Once again, substitutions are only coded with respect to words in a specified include list and only if the substitution flag is given by the user.

If the prerequisite amount of overlap exists between the source and response utterances, a routine is called to perform the following functions: (1) create coding tiers; (2) update statistics concerning additions, deletions, exact matches, and substitutions; and (3) update statistics concerning exact matches, expansions, and reductions. Note that if a word list has been included, functions 2-3 are calculated strictly with respect to the words in that list.

Summary Statistics

After the program is finished coding a file or a group of files, some descriptive statistics based on these codes are output. These statistics include absolute and proportional values for each of the coding categories mentioned earlier for each of the speaker types. The absolute values are to inform the researcher of the number of instances a certain interactional pattern has occurred. The proportional values allow the researcher to compare results across corpora of different sizes. Examples of some summary statistics are provided in Appendix C. The reader will note that the statistics are formatted to make it easy to perform additional statistical analyses. Column 3 is for the adult, and column 4 is for the child.

User-Specified Options

Three major options may be specified by the user. The first is the minimum repetition index. The user may set this value so that any source-response utterance pair below the minimum repetition index will not be coded or entered into the tabulation of summary statistics. Second, the user may specify an include file of words. If an include file is given, all summary statistics refer only to the words listed in that file. This is useful for exploring word-class differences. Third, the user may ask for substitutions to be coded. This option is only meaningful in the presence of an included word list, since substitutions are coded with respect to this list.

Selecting Corpora for Analysis

For a corpus to be useful for the basic descriptive database, it must consist primarily of parent-child interaction. Ideally, the corpus should also cover as large a longitudinal slice as possible. Several corpora in CHILDES match these requirements. Three of the larger corpora have roughly matching age ranges which will allow analyses across subjects. These include:

Adam (Brown, 1973) was the child of a minister and an elementary school teacher. His family was middle class and well-educated. Though he was Black, he was not a speaker of American Black English but of Standard American. There are 55 files in the Adam corpus and his age ranges from two years, three months to four years, ten months. A minimum of two hours a month was recorded sometimes more when development was occurring rapidly. The average was two-hour visit every second week.

Sarah (Brown, 1973) was the child of a working class family. There are 139 files in the Sarah corpus covering the ages two years, three months to five years, one month. A minimum of two hours a month was recorded sometimes more when development was occurring rapidly. The average was two-hour visit every second week.

Abe (Kuczaj, 1976) is the son of a child language researcher. The corpus consists of a diary study of a single child done by Stan Kuczaj. Approximately one hour of spontaneous speech in his home was recorded each week (two one-half hour sessions per week) from 2;4 to 4;0 with one-half hour of spontaneous speech being recorded each week from 4;0 to 5;0. There are 210 files in the corpus. This corpus contrasts with the others as being mostly father-child rather than mother-child.

There are other longitudinal corpora in CHILDES; these three were chosen because of their size and the overlap in the age ranges.

A DISTRIBUTIONAL ANALYSIS OF PARENT-CHILD CONVERSATIONAL INTERACTION

Method

Three corpora were coded and analyzed by the CHIP program. The number and proportion of adult and child operations were tabulated. For the overall analysis, the operations included the addition, deletion, and exact match

of lexical items. For the analysis of word-class differences, the substitution operation was also coded. Parent-child interaction for nine different word classes was also examined. The classes included: (1) variations of *be*; (2) variations of *do*; (3) variations of *have*; (4) modals; (5) articles; (6) subject pronouns; (7) adjectives; (8) nouns; and (9) verbs. The list of open class items (adjectives, nouns, and verbs) was generated from the 250 most frequent words in spoken child language as compiled by Hall, Nagy, and Linn (1984). These items were selected to minimize ambiguity. The exact word lists are presented in Appendix A. Nonambiguous contracted forms were also included, so that all usages of the items in questions were tabulated. For example, the unambiguous form *I* was included, but the ambiguous form *'s* was not. The three corpora of Abe, Adam, and Sarah were divided into six different age sequences: 2;0-2;6, 2;7-3;0, 3;1-3;6, 3;7-4;0, 4;1-4;6, and 4;7-5;0. Note that in all tables and figures, each age level will be referenced by the age ending the time period (i.e., 2;0-2;6 will be referred to as 2;6). Both mothers and fathers were included in the analysis, but speech from other participants was not coded.

The following examples are provided to clarify the coding categories described above. There are eight examples: additions, deletions, exact matches, and substitutions for both the parent and the child. The examples vary significantly in the presence and accessibility of relevant information for the language-learning child. The parental additions and substitutions offer the child accessible information about auxiliaries (in question formation) and pronouns. The parental deletions and exact matches are not as informative. The operations performed by the children reveal some of their abilities (or the lack thereof). In the deletion and exact match examples, the child's utterances are ill formed. These two operations are the most revealing about the status of the child. In contrast, the child addition and substitution are advanced and well formed.

Parent addition of *be*:

*ADA: And dere they go?

*MOT: And where are they going? (4)

Parent deletion of *have*:

*ABE: And I have two hands.

*MOT: Two hands for what? (5)

Parent exact match of common noun:

*SAR: I have to # play games?

*MOT: Every time you say the word.

*MOT: Now you play games with Gloria. (6)

Parent substitution of pronoun:

*ADA: I will be a duck with a hand.

*MOT: You will be a duck with a hand. (7)

Child addition of *do*:

*MOT: It's just a toy snake.

*ADA: Mom # what cha do with a toy snake? (8)

Child deletion of do:

*MOT: He doesn't have fingers.
*ADA: Doggie have a finger on too. (9)

Child exact match of common noun:

*FAT: Whose girl are you?
*SAR: Mommy girl! (10)

Child substitution of pronoun:

*MOT: You need to take a bath.
*ABE: No I don't want to take a bath. (11)

Amount of Interaction

The goal of this section is to give the reader an idea of the amount of parent-child conversational data that was successfully coded and analyzed by CHIP. Toward this goal, Table 1 lists the absolute number of coded responses for both parent and child for each of the three corpora. The combined total of 151,900 coded interactions is an extremely large dataset by any measure; it could not have been coded manually. In addition, as can be seen in Table 2, a substantial proportion of these responses contain

some overlap (at least one word) with their source utterances and are therefore coded for interactional operations. Finally, the percentage of repetition within all responses (including the nonoverlapping ones) is listed in Table 3. Interestingly, these general results provide a glimpse of things to come: Both Tables 2 and 3 reveal a developmental trend toward less repetition.

Pattern of Interaction

In this section, the pattern of interaction for parent and child speakers as a function of child age will be explored. Interactional patterns will be measured by differential usage of the operation types *addition*, *deletion*, and *exact match*. The absolute number of words involved in each of the three operation types was transformed into proportions by noting the rate of usage of each operation with respect to the other operations. In other words, if there were 30 additions, 15 deletions, and 5 exact matches for a particular speaker within an age level, the proportional measures for that cell would be 60% additions, 30% deletions, and 10% exact matches. As can be seen in Table 4, parents add more than their children, children delete more than their parents, and parents tend to repeat what their children have said slightly more than the children repeat what their parents have said.

Table 1
The Absolute Number of Coded Responses Across Age for Abe, Adam, and Sarah

		2;6	3;0	3;6	4;0	4;6	5;0	Total
Abe	Parent	803	6829	5577	4290	1871	1968	21338
	Child	752	6760	5709	4553	2048	2048	21870
Adam	Parent	2470	4640	5297	4028	2048	1131	19614
	Child	4763	7669	9022	5990	3777	2294	33515
Sarah	Parent	3242	6516	5915	5213	4639	3583	29108
	Child	2721	5719	5213	4351	4966	3485	26455
Total		14751	38133	36733	28425	19349	14509	151900

Table 2
The Percentage of Overlapping Responses Across Age for Abe, Adam, and Sarah

		2;6	3;0	3;6	4;0	4;6	5;0
Abe	Parent	0.45	0.40	0.42	0.38	0.37	0.35
	Child	0.41	0.35	0.39	0.36	0.35	0.34
Adam	Parent	0.50	0.48	0.42	0.41	0.37	0.40
	Child	0.28	0.24	0.27	0.29	0.26	0.29
Sarah	Parent	0.26	0.26	0.23	0.24	0.24	0.25
	Child	0.27	0.24	0.16	0.21	0.19	0.19

Table 3
The Percentage of Repetition Within Responses Across Age for Abe, Adam, and Sarah

		2;6	3;0	3;6	4;0	4;6	5;0
Abe	Parent	0.14	0.11	0.10	0.08	0.08	0.07
	Child	0.15	0.08	0.08	0.08	0.08	0.07
Adam	Parent	0.22	0.22	0.17	0.15	0.12	0.13
	Child	0.10	0.08	0.08	0.08	0.07	0.07
Sarah	Parent	0.12	0.11	0.09	0.09	0.08	0.08
	Child	0.11	0.11	0.06	0.07	0.06	0.05

Table 4
The Proportion of Words Involved in
Each of the Three Operation Types

Operation	Speaker	Proportion
Addition	Parent	0.44
Addition	Child	0.35
Deletion	Parent	0.35
Deletion	Child	0.47
Exact match	Parent	0.21
Exact match	Child	0.18

The developmental results are graphed in Figure 1. These data reveal that the interactional patterns vary according to age level. Parental additions decrease at about the same rate as that at which child deletions increase; conversely, child additions increase at about the same rate as that at which parent deletions increase. Finally, parent and child exact matches decrease slightly at similar rates with age.

In sum, these contingent patterns represent the input that children receive over an extremely large range of

communicative situations. The patterns should probably not be viewed as evidence of conscious strategy on the part of the parent. For the most part, they no doubt represent a tendency on the part of the parent to add material not present in the incomplete child repertoire as part of the normal conversational flow. With this many utterance pairs, we know that it is possible to differentiate conversational interaction further, on the basis of codes that could only be computed manually. Even so, some further distinctions are possible to compute automatically, and attempts will be made to do so in the following section.

Word-Class Differences

We know from the previous section that there is a strong relationship between parental and child interactional variables. In this section, we are interested in determining whether this relationship changes as a function of word class. We expect that it would, given differences in acquisitional rates and answerhood requirements. To test this hypothesis, CHIP coded and analyzed the three corpora with respect to nine different word classes (as

Figure 1. The proportion of words involved in each of the operation types for the parent-child dyads (A,a = addition, D,d = deletion, E,e = exact operation; parent is in uppercase, child is in lowercase).

Figure 2. The proportion of words involved in each of the three operation types according to word class (A,a = addition, D,d = deletion, E,e = exact operation, S,s = substitution; parent is in uppercase, child is in lowercase).

A. First and Second Person Pronouns

B. Third Person Pronouns

Figure 3. The interactional patterns for first- and second-person as compared with third-person pronouns (A,a = addition, D,d = deletion, E,e = exact operation, S,s = substitution).

described in the Method section). The proportional results are graphed in Figure 2.

The overwhelming result is that the contingent relationship between parents and their children in conversational interactions is maintained across word classes. With respect to the different word classes, the first result that stands out is the overall similarity between the closed-class verbs. Since we know from the child language literature (e.g., Brown, 1973) that there is a clear acquisitional pattern, it is not surprising to find developmental differences in additions and deletions. The clearest divergence from this trend is the greater number of substitutions for modals. This is not surprising, however, since modals form a more heterogeneous group than do the other closed-class verbs. The other two closed-class items also show the addition-deletion developmental pattern. However, the results for articles show an increase of exact matches over other operations. This becomes easy to understand when

we consider that the function of articles in conversation is to communicate definiteness, and that definiteness is not going to change within conversations. Pronouns are an interesting class; once the addition-deletion pattern levels off, there is a large increase in substitutions. This is also intuitive, since we would expect free substitutions for first- and second-person pronominals. In order to determine if this indeed was the case, two additional analyses of pronouns were performed: one for only first- and second-person pronouns, and a second for third-person pronouns. The results from this analysis indicate that substitutions become the predominant form of usage for first- and second-person pronouns for both parents and children by the age of 3;0–3;6 (see Figure 3). For the open-class items, the patterns for nouns and adjectives are similar, in that there is a large proportion of exact matches early in development that tails off as the children get older. However, the results for open-class verbs are different.

There is a large proportion of substitution operations. This is no doubt an artificial result due to the large amount of morphological variation in the word list (see Appendix A).

The results for both the full corpora and the word-class analyses indicate highly contingent differences between parent and child speakers as a function of child age and ability across different word classes. These results are consistent with other recent findings concerning parental input (Bohannon & Stanowicz, 1988; Demetras et al., 1986; Hirsh-Pasek et al., 1984) and continue to suggest that the no-negative-evidence claim is, at best, tenuous.

CONCLUSION

The present data were obtained with the aid of a new research tool: automatic data coding and analysis by computer. Automatic data coding and analysis programs have many advantages over manual coding. They have the potential to save time and effort and to reduce manual coding errors. Their ease of use greatly facilitates replication. CHIP, an example of such a tool, has provided detailed information concerning the exact nature of parent-child conversational interactions through analyses of over 150,000 utterances. To code and analyze such a large corpus of data manually would have taken a prohibitive amount of time and would have entailed a large number of coder errors due to the amount of detail involved. In addition, since CHIP not only tabulates quantitative data but also actually inserts codes into the transcripts, it is possible to perform many more analyses on these data. In fact, code insertion provides the best of both worlds, allowing the human to direct and focus the computer to search for more and more detailed patterns.

REFERENCES

- BAKER, N. D., & NELSON, K. E. (1984). Recasting and related conversational techniques for triggering syntactic advances by young children. *First Language*, 5, 3-22.
- BOHANNON, J. N., & STANOWICZ, L. (1988). The issue of negative evidence: Adult responses to children's language errors. *Developmental Psychology*, 24, 684-689.
- BROWN, R. (1973). *A first language: The early stages*. Cambridge, MA: Harvard University Press.
- BROWN, R., & HANLON, C. (1970). Derivational complexity and order of acquisition in child speech. In J. R. Hayes (Ed.), *Cognition and the development of language*. New York: Wiley.
- CHOMSKY, N. (1959). A review of Skinner's *Verbal Behavior*. *Language*, 35, 26-58.
- DEMETRAS, M., POST, K., & SNOW, C. (1986). Feedback to first-language learners. *Journal of Child Language*, 13, 275-292.
- GOLD, E. (1967). Language identification in the limit. *Information & Control*, 16, 447-474.
- HALL, W. S., NAGY, W. E., & LINN, R. (1984). *Spoken words*. Hillsdale, NJ: Erlbaum.
- HIRSH-PASEK, K., TREIMAN, R., & SCHNEIDERMAN, M. (1984). Brown & Hanlon revisited: Mother's sensitivity to ungrammatical forms. *Journal of Child Language*, 11, 81-88.
- KUCZAJ, S. (1976). *-ing, -s, and -ed: A study of the acquisition of certain verb inflections*. Unpublished doctoral dissertation, University of Minnesota.
- MACWHINNEY, B. (in press). *The CHILDES Project: Computational tools for analyzing talk*. Hillsdale, NJ: Erlbaum.
- MACWHINNEY, B., & SNOW, C. (1985). The child language data exchange system. *Journal of Child Language*, 12, 271-296.
- NELSON, K. (1977). Facilitating children's syntax acquisition. *Developmental Psychology*, 13, 101-107.
- NELSON, K. E., CARSKADDON, G., & BONVILLIAN, J. D. (1973). Syntax acquisition: Impact of experimental variation in adult verbal interaction with the child. *Child Development*, 44, 497-504.
- NELSON, K. E., DENNINGER, M. S., BONVILLIAN, J. D., KAPLAN, B. J., & BAKER, N. D. (1984). Maternal input adjustments and non-adjustments as related to children's linguistic advances and to language acquisition theories. In A. D. Pellegrini & T. D. Yawkey (Eds.), *The development of oral and written language in social contexts*. Norwood, NJ: Ablex.
- ROEPER, T., & WILLIAMS, E. (1987). *Parameter setting*. Hingham, MA: Reidel Publishing.
- SKINNER, B. F. (1957). *Verbal behavior*. New York: Appleton-Century-Crofts.
- WARREN-LEUBECKER, A., BOHANNON, J., STANOWICZ, L., & NESS, J. (1986). *New evidence about negative evidence*. Paper presented at the Teachability of Language conference, Kansas City, MO.

APPENDIX A

Word Lists for the Word-Class Analyses

ADJECTIVES: better, big, different, dumb, funny, good, great, high, little, long, new, old, own, silly, small, very.

ARTICLES: a, an, the.

BE: ain't, am, are, aren't, be, been, being, how're, i'm, is, isn't, some're, these're, they're, those're, was, wasn't, we're, were, weren't, what're, where're, who're, you're.

DO: d'you, did, didn't, do, does, doesn't, doing, don't, done, whaddya.

HAVE: ain't, could've, had, hadn't, hafta, has, hasn't, have, haven't, having, how've, i've, may've, might've, must've, should've, there've, they've, we've, what've, when've, where've, which've, whose've, why've, would've, you've.

MODALS: another'll, anybody'll, anyone'll, anything'll, both'll, can, can't, cannot, could, could've, coulda, couldn't, dare, everybody'll, everyone'll, everything'll, he'll, how'll, i'll, it'll, many'll, may, might, mine'll, mustn't, must, must've, musta, mustn't, need, nobody'll, none'll, one'll, ought, oughta, shall, she'll, should, shoulda, shouldn't, some'll, somebody'll, someone'll, something'll, that'll, there'll, these'll, they'll, this'll, those'll, we'll, what'll, when'll, where'll, which'll, who'll, whose'll, will, willya, won't, would, woulda, wouldn't, you'll.

NOUNS: book, books, boy, boys, butter, car, cars, cat, cats, day, days, den, dens, dinner, dinners, dog, dogs, game, games, girl, girls, home, homes, house, juice, juices, people, piece, pieces, room, rooms, school, schools, shirt, shirts, tape, tapes, thing, things, time, water, way, ways.

PRONOUNS: I'd, I'll, I'm, I've, anybody, anybody'd, anyone, anything, couldya, d'you, everybody, everyone, everything, everything's, he, he'd, he'll, he's, I, it, it'd, it'll, it's, naught, no-one, nobody, nobody'd, nobody's, none, nothing, nothing's, one, one's, she, she'd, she'll, she's, somebody, somebody'll, somebody's, someone, someone'll, someone's, something, they, they'd, they'll, they're, they've, we, we'd, we'll, we're, we've, willya, y'all, you, you'd, you'll, you're, you've.

VERBS: ate, bought, buy, buying, came, come, coming, eat, eaten, finish, finished, finishing, gave, get, getting, give, giving, go, goes, going, got, hear, heard, keep, keeping, kept, knew, know, knowing, let, letting, made, make, making, pick,

picked, picking, put, putting, said, sat, say, sit, sitting, take, taking, tell, telling, thank, thanked, thanking, think, thinking, thought, told, took, wait, waited, waiting, want, wanted, wanting, went.

APPENDIX B

More Examples of the CHIP Coding System

(Note: Examples 2, 6, and 7 assume an include list of pronouns.)

EXAMPLE 1:

*CHI: Big-Bird hug Kermit
 *MOT: Isn't that cute.
 %adu: \$NO__OVERLAP \$REP = 0.00
 *MOT: Big Bird is hugging Kermit.
 %adu: \$EXA:kermit \$ADD:big-bird-is-hugging
 \$DEL:big-bird-hug \$REP = 0.20

EXAMPLE 2:

*CHI: I'm sitting down.
 *MOT: That's right [#] You're sitting down.
 %adu: \$EXA:sitting-down \$ADD:that's-right \$SUB:you're
 \$REP = 0.60

EXAMPLE 3:

*CHI: No flopping around.
 *MOT: Flopping around.
 %adu: \$EXA:flopping-around \$DEL:no
 \$REDUCED__MATCH \$REP = 0.67

EXAMPLE 4:

*CHI: Mommy sleeping.
 *MOT: No [#] Mommy's not sleeping.
 %adu: \$EXA:sleeping \$ADD:no-mommy's-not
 \$DEL:mommy \$REP = 0.25

EXAMPLE 5:

*CHI: Play bridge with them.
 *FAT: You'll play bridge with them.
 %adu: \$EXA:play-bridge-with-them \$ADD:you'll
 \$EXPANDED__MATCH \$REP = 0.80

EXAMPLE 6:

*CHI: I rebroke it.

*MOT: Clumsy, you broke it again.
 %adu: \$EXA:it \$ADD:clumsy \$ADD:broke \$ADD:again
 \$DEL:rebroke \$SUB:you \$REP = 0.33

EXAMPLE 7:

*CHI: I build house.
 *MOT: Yes, you built the house.
 %adu: \$EXA:house \$ADD:yes \$ADD:built-the
 \$DEL:build \$SUB:you \$REP = 0.33

APPENDIX C

Examples of Summary Statistics for Adam: 2;0-2;6

chip -cADA -pMOT -pFAT -ns -no adam01-adam06.cha:

adam.cha	Rep_Index	0.22	0.10
adam.cha	Total Responses	2470	4763
adam.cha	Overlap	1243	1342
adam.cha	No Overlap	1227	3421
adam.cha	ADD_OPS	1452	1042
adam.cha	DEL_OPS	818	1672
adam.cha	EXA_OPS	1533	1403
adam.cha	%_ADD_OPS	0.38	0.25
adam.cha	%_DEL_OPS	0.22	0.41
adam.cha	%_EXA_OPS	0.40	0.34
adam.cha	ADD_WORD	3427	1427
adam.cha	DEL_WORD	1069	4616
adam.cha	EXA_WORD	1941	1864
adam.cha	%_ADD_WORDS	0.53	0.18
adam.cha	%_DEL_WORDS	0.17	0.58
adam.cha	%_EXA_WORDS	0.30	0.24
adam.cha	AV_WORD_ADD	2.36	1.37
adam.cha	AV_WORD_DEL	1.31	2.76
adam.cha	AV_WORD_EXA	1.27	1.33
adam.cha	EXACT_REPET	157	71
adam.cha	EXPAN_MATCH	370	69
adam.cha	REDUC_MATCH	60	365
adam.cha	%_EXACT_REPET	0.13	0.05
adam.cha	%_EXPAN_MATCH	0.30	0.05
adam.cha	%_REDUC_MATCH	0.05	0.27