ISLE Workshop on Multimodal Dialog Tagging, Dec 2002

Towards modeling user behavior in human-machine interactions: Effect of Errors and Emotions

Shrikanth Narayanan

University of Southern California – Integrated Media Systems Center
Speech Analysis and Interpretation Laboratory: http://sail.usc.edu

Abstract

Data-driven approaches to spoken dialog strategy design rely on a sound understanding and modeling of user behavior in their interaction with machines. The spoken language user-machine communication channel is inherently noisy; noise in the channel may be due to errors in machine speech recognition, language understanding or other machine/user communication uncertainty and errors. Hence, annotation of human-machine dialogs needs to pay special attention to user behavior under errors and uncertainty. Another key aspect of user behavior is the dynamics of affect or emotions during an interaction relating to "how", contra "what", information is being conveyed. The goal of our work is on developing an account of user behavior working with annotated data from real human-machine mixed initiative dialogs. We illustrate the details of preparing the data for these needs using two case studies. First, we consider the DARPA Communicator dialog corpus to examine categories of error perception, user behavior under error, effect of user strategies on error recovery, and the role of user initiative in error situations. A conditional probability model smoothed by weighted ASR error rate is proposed. Second, we consider data and tagging needs for tracking the "emotional" aspects of human-machine interaction. Toward that end, we used data from a commercially deployed airline information systems and a WoZ study of child-machine interactions. Issues related to capturing emotionally salient information at the lexical and discourse level are highlighted in the context of automatic emotion recognition.

1. Introduction

Understanding and modeling user behavior is an essential step toward modeling and optimizing human-machine spoken dialog interaction strategies. Efforts along these lines are in progress [4,5] with the recent deployment of several complex dialog systems, for e.g., [1,2,3]. It is well known that many of the underlying components of the state-of-the-art dialog systems such as automatic speech recognition (ASR) and natural language understanding (NLU) rely on data-driven statistical models and, in general, are prone to errors of varying types and extent. In addition, there are other possible system and user induced errors. Our work targets user behavior modeling under such error conditions in the context of human-machine spoken dialogs. There have been several previous studies on human-machine dialogs, of varying capabilities/complexity, focusing on detecting and responding to errorful scenarios. All these studies rely on corpus annotation for various things such as marking error occurrences, error types, user response types.

For one, these annotations are essential in correlating objective measures (e.g., acoustic-phonetic prosodic measures [11], ASR word error rates etc.) with specific user behavior patterns. In turn, such models can be exploited in devising optimal strategies that may lead to a successful interaction. Major challenges here include consistent tracking of a dynamical sequence of events that spans several turns and finding correlates (features) that are conducive to learning and automatic tracking.

Yet another requirement for annotation stems from the need for enriching the description of user behavior to include "emotions". Detecting and utilizing such meta information is again hoped to enable user adaptive dialog strategies. (Although negative emotions such as user frustration are found to be correlated with system errors, this dimension of annotation is independent in itself). The goal of an automatic emotion recognizer typically is to assign category labels that identify emotional states. While cognitive theory in psychology argues against such categorical labeling [18], it provides a pragmatic choice, especially from an 'engineering standpoint'. The primary reasons for this result from (1) A lack of a definite description and agreement on a set of basic emotions [18,19] (2) A lack of consistency in description: the same emotional category tends to be described in the literature in diverse manner [20]. While the ability to recognize a large variety of archetypal emotions - happiness, sadness, fear anger, surprise, and disgust -- is attractive, it may not be practical or necessary in the context of specific applications. Based on this assumption, we favor the notion of *application-specific* emotions and thus focus on a reduced space of emotions, in the context of developing algorithms for conversational interfaces. In the context of a conversational interface, it becomes possible to combine lexical, semantic and discourse information with acoustic speech information to contribute toward emotion recognition.

The DARPA Communicator spoken dialog systems, implemented at several sites, represent some of the most recent advances in the design of mixed-initiative spoken language systems [1,2,5]. The availability of transcripts of realistic spoken dialogs from some of those systems provides an excellent opportunity to investigate the behavior of human and machine interactions in mixed-initiative dialogs. In the first case study (Sec. 2), we set out to understand the dynamics of user behavior under system errors and how the combination of system errors and user reactions to them affect the ultimate success of a dialog. In preparation for this study, we annotated a portion of the June 2000 Communicator dialogs for several features, including a categorization of both user and system behavior [27].

In the second set of case studies (Sec 3), we explore issues related to detecting and tracking "emotional" aspects of user behavior using spoken language information. To that end, we use data from two different human-machine dialog corpora. First we explore how lexical and discourse information can be

exploited for emotion recognition using a study of a child-machine game dialog corpus [26]. Next, assuming a limited domain, we illustrate the implementation of an automatic emotion detector that combines various elements of spoken language [21,22].

2. Case study 1: Understanding user behavior under error

The data used were the orthographically-transcribed travel arrangement dialogs from the DARPA Communicator project recorded in June 2000. Each dialog consists of some number of exchanges between a computer travel agent and a human. and is represented as a three-line triple consisting of a system utterance, a user utterance (manually transcribed from recordings), and what the ASR system heard and provided as input to the dialog system.

The data and the collection procedure are described in detail in [5]. In the Communicator dialogs, 85 experimental subjects interacted with 9 different "travel agent" systems. Of the 765 possible dialogs, many are empty, or contain no user participation. We worked with about 141 of those total dialogs (that consisted of at least 1 turn). The average length of these dialogues was 18 exchanges. The amount of data is comparable to the data considered in a similar study by Aberdeen et al [6].

2.1. Tagging Dialog Data for behavior under error

Following a review of the recent work on analysis of human computer dialogs, we devised a tagging scheme consisting of 23 tags with which to monitor 3 dimensions of the dialogs: user behavior, system behavior, and task status. Since our goal was to do a quantitative analysis of the (disruptive) effect of errors, existing tagging schemes, while instructive, were not directly applicable. Automatic analysis of error conditions beyond the ASR word error rate is difficult without the aid of manual tagging. Hence, manual tagging was necessary. However, for example, unlike [6], we do not keep track of the subtask in which the error occurred, nor do we distinguish between dialog acts as in [7]. Finally, the user utterances in the communicator data are very short, averaging 3 words. Under these circumstances, we also have not made an to attempt labeling disfluencies as projects dealing with longer, more open-ended utterances have done [9][10].

The detailed tag set together with usage conventions and examples of application are provided in http://sail.usc.edu/dialog/model_tags. Briefly, the tag set for our purposes included (1) SYSTEM tags: explicit confirmation, implicit confirmation, help, system repeat, reject, non sequitur (2) USER tags: repeat, rephrase, contradict, frustrated, change request, startover, scratch, clarify, acquiesce, hang-up (3) TASK tags: error (at the recognized utterance), back on track, task success.

For error segments, we locate the beginnings of errors, and place a generic "error" tag on the ASR output that resulted in an error (Note that the standard ASR word error rate for each turn is also calculated). Within error segments we focus on three phenomena: system utterances which exhibit a system reaction to the error, user utterances which react to or try to correct the error, and the means by which the user becomes aware of the error. Sometimes the user becomes aware of an error because of a system rejection such as, *"I'm sorry, I couldn't understand you."* or a verbatim repetition of a system prompt for information. Other times implicit confirmations or non sequiturs in system utterances alert the user to the presence of an error, in which case the user must try to make the system aware of the error. Because the scenarios were conducted by paid subjects arranging for hypothetical travel for this particular data collection, some users had a tendency to acquiesce to errors that proved difficult to correct, or even to change the nature of the travel request in response to repeated recognition errors. These deviations from the original plan are also marked.

Finally, we tag the point at which the dialog gets back-on-track (BOT), marking the system utterance in which the user could reasonably discover that the portion of the task derailed by an error has been successfully understood. At the end of the dialog we indicate whether the arrangements were successfully completed or ended in a hang-up or acquiescence to some error. The tagging was done by two annotators and showed 87% inter-annotator agreement. The tagging conventions used allow the assignment of all applicable tags to the dialogs. The agreement measure used was the number of identically tagged lines, divided by the number of lines reviewed and tagged. The measure is conservative in that it counts as agreement cases where 100% identical tagging appears on exactly the same line for both annotators. It does not include partial overlap, or positional offset.

Following the tagging itself, we analyzed the dialogs and user histories from several perspectives, seeking patterns in user behavior, and correlations between user behavior and the length and severity of error segments.

2.2. Results and Discussion

Firstly it is useful to get a general sense of the presence of errors in the dialogs. The data, overall, is dominated by errors of various types. The roughly 2528 turns we tagged consists of 141 dialogs conducted with 35 paid subjects. The dialogs contain 235 error segments. Note that according to our definition an error segment can (1) end in either by getting back on track (BOT) with perhaps a complete success, acquiescence or abort (2) be nested within another error segment. Of these 235 segments, 78% got back on track.

Figure 1 provides the distribution of error segment length (number of turns) in the data. About 80% of these are between 1-9 turns with most of them between 2 to 4 turns. Of these, the average length of the error segments that eventually get back-on-track is 6.7 and those that never recover is 10. From these numbers alone, we do not know whether the length of the unrecovered errors represents something about the system or user, or if it represents some threshold of user tolerance for error resolution beyond which users will simply hang up rather than continue.

We present analysis results on the following points: (1) Categories of error perception (2) User behavior under error including user initiative in error vs. non-error situations.

2.3. Categories of error perception

Here, we see whether the manner in which the user discovered the error affects the time to get back on track. In the case of a system prompt repetition or a system rejection, the user is explicitly made aware of an "error" (from its perspective). In the case of an implicit confirmation or a system non sequitur, it is up to the user to notice that an error has occurred and draw the system's attention to this. In Table 1, we present error segments grouped by the way in which the

user becomes aware of the error, to see if the way in which the error is discovered affects the time to recover or success in recovery.

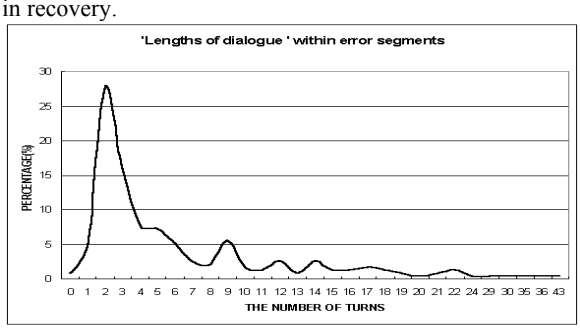

Figure 1: Normalized histogram of the length of error segments (number of turns).

We can roughly divide the error discovery types into high frequency (system rejection, implicit confirmation, & system prompt repeat), and low frequency (explicit confirmation & non-sequitur). Among the high-frequency error discovery types, it is striking that *implicit* confirmation results in a much longer time to get back on track (10 exchanges vs. 6), and a much lower rate of getting back-on-track at 68%, compared to 80% and 90% for the other high-frequency errors.

Error perception	# of err segments	avg err length for BOT	avg err length not BOT	%BOT
Reject	35	6	7.8	83%
Implicit	25	9.6	14.6	68%
Repeat	21	5.8	13	90%
Explicit	10	5.5	8.75	60%
Non-seq	9	6	7.5	77%

Table 1: Lengths of error segments which did get back-on-track (BOT) and those which didn't, as well as the percentage of errors that eventually got back on track.

2.4. User behavior under error

We next examine the distribution of user strategies in coping with errors. Fig. 2 shows the distribution on the user behavior immediately following an error (in the previous turn).

The next two tables show the distribution of user strategies for segments that eventually did get back on track and for those that never got back on track:

We observe that users in the successful error recoveries (see Table 2) use significantly ($p < 0.1$, ANOVA) more rephrasing than those in the unrecovered errors and less contradictions (e.g. "not 3 am, 3 pm") (Table 3). They also make use of the "start over" and "scratch" features more to terminate error episodes rather than trying to repair chains of errors. Users in successful error recoveries were also much more likely to work around system weaknesses by changing their travel plans! While this apparently got the dialog back on track, it is not a viable strategy for real travel arrangements.

Degree of Error and User behavior: Errors in spoken dialogs are not merely binary valued and it is critical to incorporate the degree of error into the modeling. To illuminate user behavior under error further, we considered the user response conditioned on the system strategy to estimate the probability P(User Behavior|System Behavior), P(U|S) from now on. It has been well accepted in the field that ASR word error rate (WER) is a good correlate of dialog

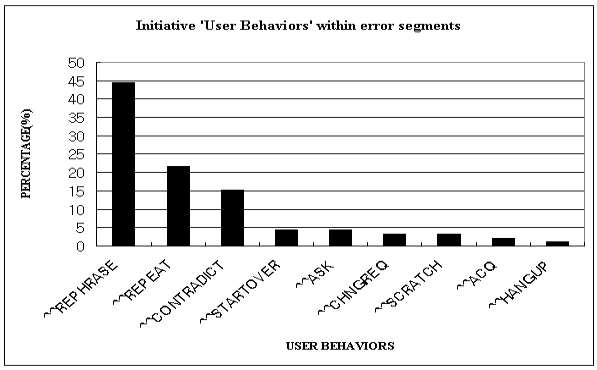

Figure 2: 'User Behavior' after the first error within an error segment. Rephrasing was the most frequent user behavior and Hang-up was the least frequent user behavior.

frequency normalized for length of errors	User strategy in Errors that got back-on-track
0.130	Repeat
0.117	Rephrase
0.077	Contradict system
0.055	Start over
0.045	Ask
0.022	*Change request*
0.015	Scratch
0.005	Acquiesce to error

Table 2: Prevalence of user strategies in error segments which eventually got back on track.

frequency normalized for length of errors	User strategy in Non-back-on-track
0.114	Repeat
0.102	Contradict system
0.071	Rephrase
0.055	*Hang up*
0.031	Start over
0.024	Ask
0.012	Scratch
0.012	Acquiesce to error
0.004	Change request

Table 3: Prevalence of user strategies in error segments which did not get back-on-track

performance [5]. Hence as a first approximation, we smoothed the probability mass of P(U|S) using an exponentially-weighted WER measure (1-10**(-WER*k/100)) that maps WER (which can be between 0 and infinity) to a range between 0 and 1. For the calculations below we chose k=1; it could vary from system to system. The results are shown in Figure 3. The most common user behavior here is rephrasing or repeating the previous request, contributing to 82% of all user responses under error. Canceling/changing the previous request or starting over are relatively rare user behaviors under error.

This is further exemplified in Fig. 4 that shows the conditional (smoothed) distribution for P(U|S=SYSTEM

REPEAT), corresponding to a highly popular system strategy when the system is "cognizant" of an error.

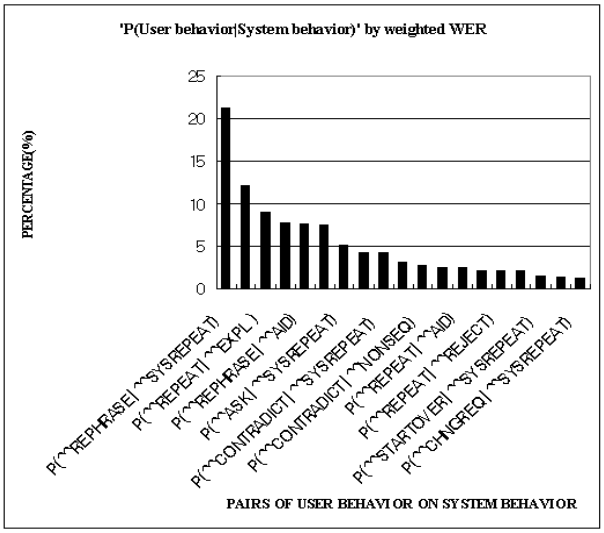

Figure 3: P (User behavior | System behavior) smoothed by exponentially weighted WER.

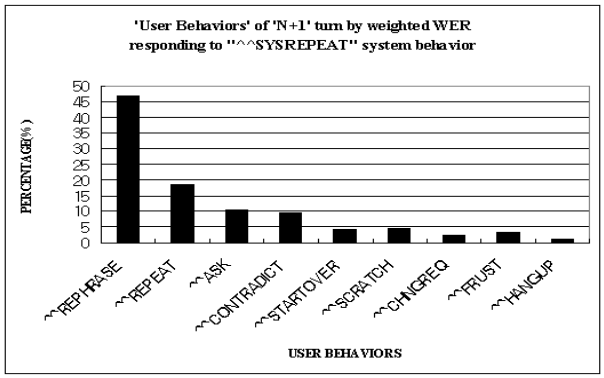

Figure 4: Smoothed Conditional Probability of User Behavior in (N+1)th turn based on weighted WER of 'SYSTEM REPEAT' system behavior in the N-th turn.

It is similarly interesting to look at user behavior when the system is not (necessarily) cognizant of an error such as when using an implicit confirmation strategy. Fig 5 shows teh smoothed distribution for P(U|S=IMPLICIT). Not surprisingly, the user is most likely to contradict the erroneous system behavior.

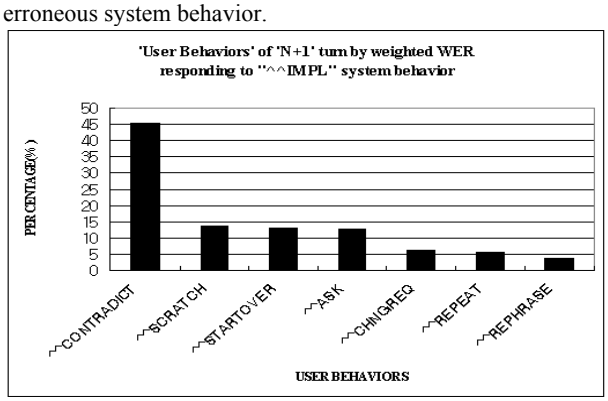

Figure 5: Smoothed Conditional Probability for User Behavior in (N+1)th turn based on weighted WER of 'IMPLICIT CONFIRM' system behavior in the N-th turn..

User initiative in error and non-error environments: Here we look at the user's tendency to use initiative over the course of the dialog. We have considered user initiative to be the cases where the user did not simply respond to system prompts, but attempted to guide the dialog themselves. The one part of the dialog that often looks the most like user initiative (and which often fails) is the response to the open prompt at the beginning of most of the dialogs. However, since this is a free-form answer to an open question, we have not tagged it as initiative. It is clear from Table 4 that user initiative behavior is significantly more in error segments than not ($p < 0.05$).

User Initiative tag	Frequency in error segments	Frequency in non-error segments
Ask	0.0319	0.0060
Contradict	0.0707	0.0121
General initiative	0.1647	0.0424

Table4: Frequency is normalized over all dialogs.

3. Case study 2: Tracking User Emotions from Spoken Language

Here, we focus on learning the "emotional" aspects of user behavior from spoken language data. Most of the reported studies have used speech recorded from actors that were asked to express (feign) pre-defined emotions; furthermore, they have predominantly focused on using just the acoustic information from the speech signal. Finally, a majority of these data were isolated utterances i.e., not obtained in a dialog context. One notable exception is the study by Batliner et al. [23]. In this work, a 'Wizard-of-Oz' scenario was used to collect data. Subjects were assumed naive and supposed that they were communicating with a real computer, and the study reported classification of the utterances into two emotions: 'emotional' and 'neutral'. The authors used details about topic repetition as their 'discourse' information to improve emotion recognition accuracy. Another related study that used language and discourse information, more in the context of identifying problematic dialogs is one by Langkilde et al [10]. More recently, a study by Ang et al [24] has explored the detection of user frustration using a number of dialog level features based on the DARPA Communicator corpus.

In this section, we explore the detection of user emotions using two human-machine dialog corpora. First, to explore the role of lexical and discourse information in emotion recognition, we detail an analysis of discourse markers related to frustration and politeness (for example, the use of swear words, negation, and the repetition of the same sub-dialog) using a child-machine interaction corpus [13,26]. Second, we use a corpus of utterances obtained from a commercially deployed human-machine spoken dialog to show some preliminary results in implementing an automatic emotion recognizer. The availability of a constrained-domain dialog application, such as automated call center, provides the possibility of utilizing spoken language information along a number of dimensions such through the use of acoustic, lexical (word choices) and discourse correlates of emotions.

Constraining the domain of emotions makes annotation of basic affect categories feasible and enables the possibility of applying learning algorithms.

3.1. Case study of a child-machine dialog game

This case study explored a child-machine dialog corpus for gleaning linguistic patterns related to user emotions. One hope was that an application-dependent "dictionary-driven" approach can be designed (eventually) to automatically categorize user emotions. (The premise for the child-machine interaction study was that computer systems interacting with children need to be tailored for these users so that they will understand child intent and so that the child will have a positive and successful experience with the system [12,13,17]. Children are still learning linguistic rules of social and conversational interaction. Their concepts of social structure are still solidifying and are different from those of adults. This means that their behavior in interacting with a computer as an interlocutor is also different from the behavior of adults.)

The study examined two sociolinguistic markers. The first part explored what type of vocabulary might indicate that a child is having difficulty with a task. It focuses on children's use of verbal expressions of frustration, annoyance, and rejection. In discourse between children and computers, frustration markers are expected to differ from those used in discourse between adults and computers. For example, children are expected to use less profanity, but to express frustration more often. Additionally, it is hypothesized that the use of repetition will correlate with the use of frustration markers such that when a child is experiencing difficulty with a task they will have to repeat some of their information requests. In particular, we examine differences as a function of age, sex, and task abilities.

The second section of the paper investigates how politeness and the linguistic form of information requests differ among children of different ages and sexes. Research in language acquisition shows that even six and seven year-old children have awareness and command of varying levels of politeness associated with different registers [15]. We examined the politeness of children's requests for information or action, and the register variation sensitive to the relationship between interlocutors.

3.1.1. Data and Methods

The data corpus being examined came from a 1997 study on child/machine interactions: ChIMP—*Children's Interactive Multimedia Project* database [12,13]. The total database included spoken interactions of 160 boys and girls, six to fourteen years of age, with a computer. The study used a Wizard of Oz (WOZ) design, in which a human operator controlled the computer without the knowledge of the subject. The WOZ design ensured that computer language understanding and speech recognition components of the task could be performed without error. The task was to play "*Where in the USA is Carmen Sandiego"*, a computer game familiar to many children in the United States. Text transcripts of the children's utterances were analyzed.

For the purpose of initial quantitative analysis, any speech utterance that triggered no valid game response or action was defined to be *extraneous* i.e., out of domain, primarily from the perspective of automatic speech recognition. In the data, such extraneous speech utterances corresponded to approximately 5% of all utterances spoken for the 8-10 year-olds (compared to 3.7% for all subjects), with values ranging from 0% to 25% among individual subjects (7% variance). Most extraneous speech utterances fell in one of the following categories:(i) those expressing excitement or disappointment when vital/useless information was provided by the game or success/failure was achieved in one of the game stages, (ii) those requesting game-strategy information, interpretation of game output or approval by other people in the room (an adult moderator or other children were present in the game room for about half of games played), and (iii) interacting with characters on the screen irrelevant to game goals and objectives. Overall, the extraneous speech utterances were found to be highly speaker-dependent, age-dependent, and to be preceded by a small subset of dialog states. These results motivated us to systematically investigate the importance of the linguistic patterns, initially deemed to be extraneous, to better understand child-machine interactions.

3.1.1.1 Frustration and Rejection Language

Frustration Markers

In order to determine the types of words children use to express frustration and politeness, we created a catalogued lexicon of the words found in the database. We identified 21 words likely to indicate frustration, difficulty, or annoyance. "Shut up" is the most popular frustration marker, well ahead of others such as "oh man," "hurry" (or "hurry up"), "oops," and "heck." It should also be noted that there were large individual differences across children in terms of which frustration vocabulary they used. The most extreme example is "dick" which was only said by one subject (see the example dialog below).

Counts of the occurrences of the frustration markers were compared by gender, age and game outcome (win/loss). These rates are expressed as percentages because they represent the distribution across turns containing frustration markers in a game, i.e., the number of frustration markers in one game divided by the total number of turns in that game. Males used frustration markers four times more often than females (0.16% & 0.4% respectively). Additionally, the youngest children used more frustration markers than the older children. The small sample of adults also recorded as part of the experiment also showed a large frequency of frustration markers (.29%), such than young children and adults were comparable on this measure. Finally, verbal expressions of frustration occurred more than twice as often in games that ended up in a loss than in those which were won (0.13% & 0.06% respectively). These results are quantified in the tables below (Tables 5, 6).

Table 5: Frustration marker use grouped by sex and age

female	male	8-9 y/o	10-11 y/o	12-14 y/o
0.04%	0.16%	0.22%	0.04%	0.07%

Table 6: Frustration marker use grouped by game data

game 1	game 2	won	lost
0.08%	0.15%	0.06%	0.13%

Rejection Language—"no"

We hypothesize that since young children have less developed abilities to express complex information requests, they may have a more frequent need to reject responses to their commands. Since the game scenario did not involve asking the children any yes/no questions in solving the game, the text transcripts could be examined for occurrences of the word "no" from the children as this word systematically indicated a rejection of a system response or action. Rejection using the word "no" usually happened when the child said she wanted something and then changed her mind after the computer had begun that action. The table below shows averages over the first and second game played by the child and whether the game was won or lost; both occurrences and percent occurrences per turn are given. Losing games included more use of "no" than winning games, and second games in a series had more occurrences than first games.

The table below shows occurrences of "no" as a function of child age and sex. Results indicate that males reject more than females and that the youngest children make more frequent rejections than the older children.

There are at least two possible interpretations of the sex difference. Female children might reject system actions/responses less often because they are more patient or because their information requests have been created more successfully. Alternatively, female children might in fact show comparable rejection rates to the male children but simply be using some other verbal form to do so. The age effect is also of interest. It provides initial support for our hypothesis that the information-request format of the game might create a situation in which the less developed cognitive skills of the youngest children put them in the position of having to more frequently reject system responses to the requests they've formulated.

Table 7: Occurrences of the rejection word "no"

	game 1	game 2		Won	Lost
occurrences	1.05	1.28		1.00	1.28
turns	191	154		184	162
occurrences per turn	0.47%	0.70%		0.55%	0.62%

Table 8: Occurrences of the rejection word "no"

	8-9 y/o	10-11 y/o	12-14 y/o	female	male
occurrences	1.70	0.77	0.64	0.67	1.36
turns	196	166	144	142	195
occurrences per turn	0.74 %	0.40%	0.47%	0.42%	0.60%

3.1.2. Politeness and Form of Information Requests

Different varieties of language are warranted by different social situations; such varieties are known as registers. Linguists find that different types of information requests (i.e. different registers) occur depending on the relative social standing of interlocutors. Most of the children's speech in the game consisted of requests to the computer for information or action. We examined these requests to determine the level of politeness children used when interacting with the computer. Analysis of the information requests used by the children can inform as to what social standing children assign to their computer-interlocutor.

Politeness Markers

The transcripts of the child-machine dialogues were searched for *please*, *thank you/thanks*, and *excuse me*. For the first analysis the children were divided into three groups: the youngest aged 6-8 years, the middle group aged 10-11, and the oldest group aged 13-14. A two-factor ANOVA with the independent variables of age group and sex indicates no significant effects on frequency of these terms. The means and standard deviation data are shown in the table below.

Table 9: Politeness markers across age groups

	Younger (6-8 yrs)		Middle (10-11 yrs)		Older (13-14 yrs)	
	M	*SD*	*M*	*SD*	*M*	*SD*
Thanks/Thank you	.08	.067	.093	.066	.068	.075
Excuse me	.004	.019	.022	.048	.025	.047
Please	.039	.086	.039	.072	.055	.123

However a more fine-grained examination indicates that these age grouping may be too broad. The two six-year olds were highly variable in their behavior. When they are excluded, the following patterns of variation emerges across the ages *(age 7, n=2; age 8, n=36; age 9, n=47; age 10, n=42; age11, n=37; age 12, n=28; age 13, n=24; age 14, n=13).*

While little age-related variation is seen for *excuse me*, the use of *please* and *thanks/thank you* seems to vary with age. *Thanks/thank you* is most common among the youngest and the middle ages, with the oldest children showing more variability in their usage. This pattern is repeated for *please* except that its occurrence is lacking for the seven year olds. When one looks at the overall usage of these terms in the figure below, a pattern of increasing use with age *and* increasing variability with age is apparent.

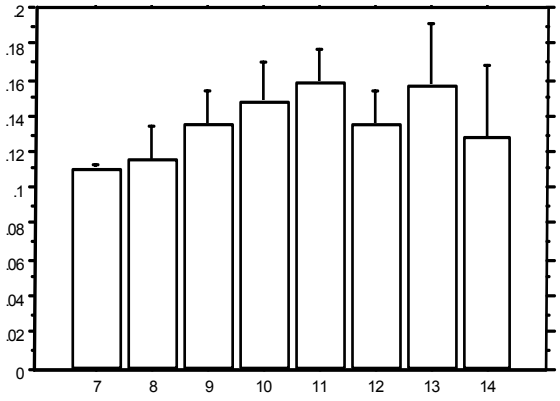

Figure 6: Cummulative use of politeness terms as a function of age.

The use of politeness markers in interacting with the computer is least common in the youngest age group, suggesting that perhaps these children preserve their cognitive resources for negotiating the game. The middle age group seems to productively use politeness markers in interacting with their interlocutor, attributing it a higher social standing than their own. The older children are particularly variable in whether they choose to use overt politeness markers. Some older children may not view their interlocutor as 'animate,' therefore not requiring extremely polite language, or they may view it as more of a peer than an interlocutor with higher social standing.

Information Requests Analysis

Another way of indicating politeness is to use different forms of questions for information or action requests. The following five forms range in politeness, from least to most polite:

- *Can you go to the map?*
- *Will you go to the map?*
- *Could you go to the map?*
- *Would you go to the map?*
- *May I see the map?*

These question forms are called modals. The transcripts were analyzed to see if use of these modal types varied by gender and/or age. The modal types were separated into the less polite (*can* & *will*) and the more polite (*could*, *would*, & *may*). The older children use significantly more of the more polite forms (M = .06, SD = .13) than the younger group (M = .01, SD = .03), with the middle group in between (M = .03, SD = .05). There were no significant differences across age groups for the less polite forms. Thus modal politeness increases with age. There were no significant differences between males and females in modal politeness. The following is a typical example from an 8-year-old male's speech:

Child: **turn left**
System: {character in focus turns left}
Child: **hi there you**
System: Hello. Nice to see you around these parts.
Child: **Can you tell me what she was wearing?**
System:…..
Child: **okay thank you**

The child uses the modal *can*, as well as overt politeness markers. Below is an example from a 14 year-old female, who uses overt politeness markers, more interrogatives and more polite modal types.

Child: **can I talk to you please?**
System: hi there.
Child: **Do you know where the suspect went?**
System:….
Child: **could you put that in my notebook?**
System:….
Child: could I look at the book?

The results suggest that the younger children don't yet have the language development and cognitive resources for politeness markers and complex forms of information requests. The preadolescent children uses overt politeness markers but don't yet fully employ polite request forms. The older children express politeness by information request forms as well as overt politeness markers, but they don't always 'deign' to use overt markers with the computer.

3.2. Automatic detection of negative emotions in a call center dialog application

The study in the previous section highlights the types of information one could glean from spoken language to help track user behavior. The case study in this explores implementing automatic detection of domain specific emotions using language and discourse information in conjunction with acoustic correlates of emotions in the speech signal. The specific focus is on detecting two emotion classes, negative and non-negative, using spoken language data obtained from a call center application. Most previous studies in emotion recognition have focused on acoustic information in speech although it is well known that language and discourse information also convey emotions. In this study, the combination of the three sources of information - acoustic, lexical and discourse - is posed as a data fusion problem to obtain the combined decision. To capture emotional information at the language level, the information-theoretic notion of 'emotional salience' is introduced. Optimization of the acoustic correlates, with respect to classification error, was done by investigating different feature sets obtained from feature selection, followed by principal component analysis.
Given the complexity in the definition and range of emotion categories, the problem related to data concerns how to obtain the required amount of realistic data to do research that yields meaningful results and algorithms. Most studies in emotion recognition in speech have used actors' voices; i.e., actors are asked to read/speak given sentences, that are usually designed to have emotionally-neutral semantic contents, with pre-specified emotions. Since those data sets are limited to short isolated utterances for archetypal emotions, results based on them may not be generalized to human-machine interaction scenarios. On the other hand, real data suffers from potential coverage problem i.e., we need vast amounts of data characterizing various emotion types, and from a number of users and contexts, to design valid models and algorithms. Our limited-domain approach allows in-depth focus on a finite set of emotions using significant amounts of data obtained from realistic human-machine interactions.

3.2.1. Data and Methods

The speech data used in the experiments were obtained from real users engaged in spoken dialog with a machine agent over the telephone for a call center application deployed by SpeechWorks International. The speech database used for our experiments contained 1187 calls, each having an average of 6 utterances; the total number of utterances was approximately 7200.

The original usage data corpus comprised calls in the order of thousands with only a small fraction representing potentially negative emotions of interest to this work. Hence, this required some pre-processing to narrow down data of interest for emotion recognition. The first step in data processing was to mine this data using semi-automatic methods using

objective measures such as ASR accuracy, total number of dialog turns, and rejection rate to narrow down the inventory to potentially useful dialogs for our experiments. This was followed by subjective tagging of the data into one of two possible emotion categories - negative, and non-negative - by four different human listeners. In our study, negative emotions represent anger and frustration in human speech, whereas non-negative emotions are the complement of that, i.e., they represent neutral or positive emotions such as happiness or delight. The order of utterances was randomly chosen in order for listeners not to be influenced in guessing the emotions by the situation in the dialogs (minimizing thus the effect of discourse context). After the human listening tests, it turned out that most non-negative emotion utterances were neutral in nature, i.e., they had no apparent display of emotions.

To measure the agreement among the taggers, the kappa statistic was used [25]. Kappa statistic provides a measure of agreement for categorical variables in subjective tests. The kappa coefficient, *K*, is the ratio of the proportion of times that the coders/taggers agree (corrected for chance agreement) to the maximum proportion of times that the coders could agree:

$$K = \{P(A) - P(E)\}/\{1 - P(E)\}$$

where *P(A)* is the proportion of times that the *k* coders agree and *P(E)* is the proportion of times we would expect the *k* coders to agree by chance. If there is complete agreement among the coders, then $K = 1$; whereas if there is no agreement (other than the agreement which would be expected to occur by change) among the coders, then $K=0$.

The results of the values of kappa statistic for female and male data were 0.45 and 0.48, respectively. It represents only a moderate agreement among the taggers. To see whether these results represent a significant difference from 0; i.e., the agreements by chance, we did hypothesis test. The results showed that they exceed the $\alpha=0.25$ significance level (where $Z = 1.96$). Therefore, we can see that the tagging exhibits significant difference from the agreements by chance.

3.2.2. Emotion Recognition Results

For the acoustic features, we started with an exhaustive list of measures suggested in the literature and pared down to obtain a rank ordered list by feature selection methods [21]. The best 15 feature set included: Ratio of duration of voiced and unvoiced region, energy mean, energy median, energy standard deviation, F0 regression coefficient, F0 median, energy regression coefficient, energy max, energy min, energy range, duration of the longest voiced speech, F0 mean, BW1, F0 max, and BW2. Optimization of the acoustic correlates, with respect to classification error, was done by investigating different feature sets obtained from feature selection, followed by principal component analysis.

To capture emotional information at the language level, the information-theoretic notion of 'emotional salience' was introduced [22]. Emotional salience is a measure of the amount of information that a specific word contains about the emotion category. A salient word with respect to a emotion category is one which appears more often in that category than in other parts of the corpus and is considered as a distance measure from the null words of which the relative frequency in each class is the same. We used the salience measure to find and associate words that are related to emotions in the speech data.

Discourse information in human-computer interaction has been suggested as being potentially useful for emotion recognition and has been combined with acoustic information to improve the performance of emotion recognizers [23,24]. In our study, the discourse labels are based on categorization of users' response. Five labels -- 'rejection', 'repeat', 'rephrase', 'ask-start over', and 'non of the above' - were used. Most of utterances were labeled as 'none of the above' and they were mostly utterances corresponding to user responses to specific information requests. As expected, a large portion of utterances in the negative emotion category had been labeled as rejection (26% for male data and 34% for female utterances). Whereas, only about 2% of the non-negative utterances were labeled as 'rejection'. As features to classifiers, we combined 'repeat' and 'rephrase' labels into a single category because they were user responses in similar situations, and helped reduce the dimension of the feature set.

A decision level fusion was used to combine the classification (linear discriminant method) results using the three aforementioned sources of information. Experimental results (Figure 7) on the call center data show that the best results are obtained when acoustic and language information are combined; while adding discourse information did not improve the overall performance, it may merely be a limitation of the application domain considered.

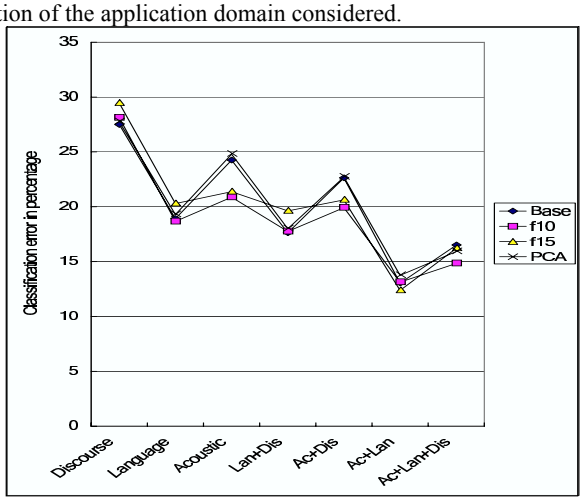

Figure 7:Comparison of various combinations of input features. Ac = acoustic correlates, Lan = language information, Dis = discourse information. Four sets of acoustic features are considered. Linear discriminant classifiers were used for classification using each information source.

If we are able to constrain in application (context) specific ways, the problem of emotion recognition becomes more tractable and can yield useful information about a user engaged in a dialog with a machine.

4. Summary

Modeling user behavior is one of the most challenging problems in spoken dialog systems research. Empirical analysis and modeling using real user data helps to illuminate

user behavior patterns. The analysis reported represents a preliminary attempt at understanding user behavior under error and uncertainty in spoken dialogs.

Results show that users discovering errors through implicit confirmations are less likely to get back on track (or succeed) and take a longer time in doing so than other forms of error discovery such as system reject and reprompts. Further successful user error-recovery strategies included more rephrasing, less contradicting, and a tendency to terminate error episodes (cancel and startover) than to attempt at repairing a chain of errors.

The most frequent user behavior to get back on track from error segments when the system signals errors is to "rephrase" and "repeat". When a user discovers an error, say through an implicit confirmation, the user tends to "contradict" or "cancel" the action rather than "rephrase" and "repeat".

There are many open and confounding issues. One key issue relates to incorporating user behavior priors (i.e., probabilities) in the model. For example, we observe that some users seem better able to avoid and/or get out of trouble. The authors of [1] observe that in this specific experimental setup, where the subjects were paid participants with no real stake in successful task completion, some users were simply inattentive or careless. In the process of tagging the transcribed data, we additionally observed that some participants had much more trouble than others getting usable ASR output. Table 5 looks at some users who participated in 5 or more scenarios.

In Table 5, two users, A and B, seem particularly successful. Although they appear to have higher numbers of errors per dialog, this is probably because they did not give up, since they also have the highest rates of recovery with relatively short error episodes. Two other users, C and D, seem the least successful. D has a very low percentage of back-on-track errors, and C seems to experience inordinately long error episodes. When we looked at the strategies these users adopted under error we found that all users tried repeating themselves. However, the less successful users frequently hung up on the dialog or started the dialog sequence over; something that the successful users were less likely to do.

User ID	# of dials	Errs / dial.	%BOT	Avg length of err segmt
1	9	1.4	.69	8.5
2	9	1.4	.76	8.9
A	8	2.9	.87	7.8
B	8	2.4	.74	4.9
C	5	1.0	.60	10.2
D	5	1.4	.42	6.0

Table5: Error-proneness in users: % BOT is the percentage of error episodes that got back on track.

These types of prior user information needs to be learnt and incorporated into the models. Ongoing work focuses on those questions and how a user model interacts with a system model in an optimization framework.

Automatic recognition of emotions from human speech, and/or other sensory modalities, by machines is gaining increasing attention from the engineering community to enrich the description of user behavior. The performance by a computer, and the emotional categories it can cover, are far limited compared with those capable by humans. One main difficulty comes from the fact that there is lack of complete understanding of emotions in human minds, including lack of agreement among psychological researchers, a pre-requisite to be satisfied in attempting to build an effective machine for the task of emotion recognition. However, we believe that we can design algorithms performing reasonably well in constrained domain-specific applications, such as automated call center application we focused on in this paper, and the knowledge gained from these efforts can help us understand deeper issues and potentially extend to more general applications. Of course, such efforts still need to face and overcome a number of challenges including dealing with issues of data sparsity, and consistent tagging to yield robust and useful models.

Acknowledgements: The work presented in this paper is a result of collaboration with a number of people including: Elaine Andersen, Dani Byrd, Alex Potamianos, Roberto Pieraccini; Graduate students: Laurie Gerber, Chul Min Lee, and Jongho Shin; Undergraduate research students: Sudha Arunachalam, Dylan Gould, and Abe Kazemzadeh all of whom contributed significantly to various aspects of these projects. Work supported in part by the Integrated Media Systems Center, a National Science Foundation Engineering Research Center, Cooperative Agreement No. EEC-9529152 and the Department of the Army under contract number DAAD 19-99-D-0046.

5. References

[1] Wayne Ward and Bryan Pellom, "The CU communicator System", IEEE ASRU, pp. 341-344, 1999.

[2] Levin, E., Narayanan, S., Pieraccini, R., Biatov, K., Bocchieri, E., Di Fabbrizio, G., Eckert, W., Lee, S., Pokrovsky, A., Rahim, M., Ruscitti, P., and Walker, M. (2000), *The AT&T-DARPA Communicator mixed-initiative spoken dialog system*, Proc. of ICSLP, (Beijing, China), pp. 122-125.

[3] V. Zue, S. Seneff, J. Glass, J. Polifroni, C. Pao, T. Hazen, and L. Hetherington, "JUPITER: A Telephone-based Conversational Interface for Weather Information", IEEE Trans. Speech and Audio Proc., pp. 85-96, 2000.

[4] E. Levin, R. Pieraccini, W. Eckert, "A Stochastic Model of human-machine interaction for learning dialog strategies", IEEE Trans. Speech and Audio Proc., 2000.

[5] Walker, M., Aberdeen, J., Boland, J., Bratt, E., Garofolo, J., Hirschman, L., Le, A., Lee, S., Narayanan, S., Papineni, K., Pellom, B., Polifroni, J., Potamianos, A., Prabhu, P., Rudnicky, A., Sanders, G., Seneff, S., Stallard, D., and Whittaker, S., "DARPA Communicator Dialog Travel Planning Systems: The June 2000 Data Collection"., Proc. Eurospeech 2001.

[6] Aberdeen, J., Doran, C., Damianos, L., Bayer, S., and Hirschman, L., "Finding Errors Automatically in Semantically Tagged Dialogues", in *Proceedings of HLT*\. 173-178. 2001.

[7] Walker, M. and Passoneau, R. "Dialog Act Tags as Qualitative Dialog Metrics for Spoken Dialog Systems". in *Proceedings of, HLT 2001*, 2001.

[8] Levow, G.-A. "Characterizing and recognizing spoken corrections in human-computer dialogue." In *Proceedings of COLING/ACL-98,* 1998.

[9] Allen, J., and Core, M. “Draft of DAMSL: Dialog Act Markup in Several Layers”. October, 1997.
[10] Langkilde, I, Walker, M., Wright, J., Gorin, A., Litman, D., “Automatic Prediction of Problematic Human-Computer Dialogues in “How May I Help You”. In ASRU, 1999.
[11] Swerts, M., Litman. D., and Hirshberg, H. “Corrections in spoken dialogue systems”, Proc. ICSLP, 2000.
[12] A. Potamianos and S. Narayanan., “Spoken dialog systems for children”, *Proc. of ICASSP*, (Seattle, Wa), p 197-200, 1998.
[13] S. Narayanan and A. Potamianos, "Creating conversational interfaces for children.", *IEEE Trans. Speech and Audio Processing*., vol 10, March 2002.
[14] E. Andersen., *Speaking with style: The Sociolinguistics Skills of Children*. London: Routledge, 1991.
[15] E. Andersen, M. Brizuela, B. DuPuy, and L. Gonnerman., “Cross-linguistic evidence for the early acquisition of discourse markers as register variable”, *Journal of Pragmatics, 31*, p. 1339-1351, 1999.
[16] D. Byrd. 1994. Relations of sex and dialect to reduction. *Speech Communication*, 15:39-54.
[17] S. Oviatt, “Talking to thimble jellies: Children’s conversational speech with animated characters”, Proc. ICSLP (Beijing, China), pp. 67—70, 2000.
[18] A. Ortony, G. Clore and J. Taylor, "The cognitive structure of emotions", Cambridge Univ. Press, 1988.
[19] R. Pluchik, "The Psychology and Biology of Emotion", Harper Collins, New York, 1994.
[20] R. Cowie et al, "Emotion recognition in human computer interaction", IEEE Signal Proc Magazine, pp. 32-80, 2001.
[21] C. Lee, S. Narayanan, R. Pieraccin, "Recognition of negative emotions from the speech signal", Proc. ASRU, 2001.
[22] C. Lee, S. Narayanan, R. Pieraccini, "Combining acoustic and linguistic information for emotion recognition", Proc. ICSLP, 2002.
[23] A. Batliner et al, "Desperately seeking emotions: Actors, wizards and human beings", Proc. ISCA Workshop on Speech and Emotion, 2000.
[24] J. Ang et al, "Prosody based automatic detection of annoyance and frustration in human computer dialog", Proc. ICSLP, 2002.
[25] J.Carletta, “Assessing agreement on classifications tasks: the kappa statistic”, Computational Linguistics, vol. 22, 1996.
[26] Sudha Arunachalam, Dylan Gould, Elaine Andersen, Dani Byrd and Shrikanth S. Narayanan, ``Politeness and frustration language in child-machine interactions", in Proc. Eurospeech, (Aalborg, Denmark), pp. 2675-2678, 2001.
[27] Jongho Shin, Shrikanth Narayanan, Laurie Gerber, Abe Kazemzadeh and Dani Byrd, ``Analysis of user behavior under error conditions in spoken dialogs", Proc. of ICSLP, (Denver, CO), 2002.

6. Appendix 1: Tag-set and Guidelines, User modeling with Communicator Data

Examples of utterances that would receive each tag are provided in the tables where possible.

System Tags: clues by which the user becomes aware of an error.

Tag “system said:” line inside and outside of error segments wherever the phenomena occurs. Multiple tags are okay.

question:	Normal question with no tags.
	And on what date didja wanna fly?
expl:	Explicit confirmation. User is asked to confirm certain input.
	Was the arrival city wichita or london ?
impl:	Implicit confirmation. The system repeats the user’s last input to introduce the following prompt.
	A flight from miami. Where do you want to go?
reject:	Rejection. The system tells the user that the recognizer either did not hear or did not understand the last input.
	Sorry, I misunderstood. Please say the name of the city or airport you wish to depart from. (reject + aid)
aid	Aid. The system instructs the user to give the input in a certain way. Often used with ^^reject and also often used in the first system prompt.
	Try asking for flights between two major cities.
nonseq	Non Sequitur. An inappropriate system response gives the user evidence of error.
	User said: [throat_clearing] *I would like to book a flight from Columbus Ohio to Phoenix Arizona to arrive before six p. m. on October fifth* **System said**: *Where are you departing from?* (= *nonseq)*
sysrepeat	System Repeat. The system repeats its last prompt.

	(a pathological example where ASR looks ok, but response is not accepted) **System said:** *What is your destination?* **User said**: *Phoenix Arizona* **Recognizer heard**: *Phoenix Arizona* **System said**: *What is your destination?* **User said**: *the destination is Phoenix Arizona* (= user rephrase) **Recognizer heard**: *The destination is Phoenix Arizona* **System said**: *What is your destination ?*

User Tags: User's response to errors.

Tag "user said" lines inside and outside of error segments wherever the phenomena occurs. Multiple tags are okay.

Information	Normal response with no tags. User just give an information to the question of the system.
repeat:	Repetition. User repeats exactly what they said in the previous turn.
rephrase:	Rephrase. User rephrases the last input, modifying choice of words, their order, etc.
	System said: *What time would you like to depart?* **User said**: *early* **Recognizer heard**: *early* (=err) **System said**: *Sorry, I misunderstood. Please give the approximate time you would like to depart*. (=reject + aid) **User said**: *eight a. m*. (=rephrase) **Recognizer heard**: *eight a m*
contradict	Contradiction. The user contradicts the system, often as a barge-in.
	System said: *What time do you want to leave phoenix* (=impl) **User said**: *no I don't want to leave Phoenix I'm starting from Columbus Ohio*
frust	Frustration. The user shows signs of anger, contempt, disgust, and frustration.
	Oh my god [uh] can we start over (=frust + startover)
chngreq	Change Request. The user tries different dates, different cities in the same state/country in an attempt to circumvent an error.
	System said: *Flying from Dulles. What city are you flying to?* **User said**: *Hilton Head South Carolina* **System said**: *At the Hilton.. What city are you flying to?* (=nonseq + sysrepeat) **User said**: *Hilton Head South Carolina* **System said**: *At the Hilton.. What city are you flying to?* (=sysrepeat) **User said**: *Savannah Georgia* (=chngreq)
startover	Start Over. The user has the system start over from scratch using the "start over" command used by most of the systems.
	System said: *Sorry, I didn't understand that. What city are you flying to?* (=reject + sysrepeat) **User said**: *START OVER*
ask	Ask. The user directs a question to the system or asks for help.
	Do I have to fly to Rome to get to Berlin ?
acq	Acquiescence. The user continues a dialogue with out trying to correct errors. May end an error segment without getting back on track (in this case it may not be on an "user said" line). When there is an ^^acq, add a note about what was acquiesced to.
hangup	Hang Up. The user hangs up in response to an error (if it is the computer that hung up, place tag on the "system said" line).

Task Tags:

Tags about the state of system/user interaction

^^err	Error. Placed at the "Recognizer heard:" turn where the initial error occurred. Ignore the requests for ID numbers at the beginning of certain dialogues. When the error is minor (e.g. late afternoon instead of early afternoon) and the user doesn't try to correct it, use ^^acq instead.

^^bot	Back On Track. The user and system successfully negotiated the correction of an error. Placed on the system said prompt that provides the user (and tagger) with evidence of being totally back on track (at the end of an error segment—never in nested errors).
^^succ	Success. The user got the tickets s/he wanted. If there are small errors like flight time, then use with ^^acq.