Characterizing and Predicting Corrections in Spoken Dialogue Systems

Diane Litman*
University of Pittsburgh

Julia Hirschberg†
Columbia University

Marc Swerts‡
Tilburg University

This article focuses on the analysis and prediction of corrections, defined as turns where a user tries to correct a prior error made by a spoken dialogue system. We describe our labeling procedure of various corrections types and statistical analyses of their features in a corpus collected from a train information spoken dialogue system. We then present results of machine-learning experiments designed to identify user corrections of speech recognition errors. We investigate the predictive power of features automatically computable from the prosody of the turn, the speech recognition process, experimental conditions, and the dialogue history. Our best-performing features reduce classification error from baselines of 25.70–28.99% to 15.72%.

1. Introduction

Compared to many other systems, spoken dialogue systems (SDS) tend to have more difficulties in correctly interpreting user input. Whereas a car normally goes left if the driver turns the steering wheel in that direction or a vacuum cleaner starts working if one pushes the on button, interactions between a user and a spoken dialogue system are often hampered by mismatches between the action intended by the user and the action executed by the system. Such mismatches are mainly due to errors in the Automatic Speech Recognition (ASR) and/or the Natural Language Understanding (NLU) component of these systems; they can also be due to wrong default assumptions of the system or the fact that a user asks out-of-topic questions for which the system was not designed. To solve these mismatches, users often have to put considerable effort into trying to make it clear to the system that there was a problem, and trying to correct it by reentering misrecognized or misinterpreted information. Previous research has already brought to light that it is not always easy for users to determine whether their intended actions were carried out correctly or not, in particular when the dialogue system does not give appropriate feedback about its internal representation at the right moment. In addition, users' corrections may miss their goal because corrections themselves are more difficult for the system to recognize and interpret correctly, which may lead to so-called cyclic (or spiral) errors.

* E-mail: litman@cs.pitt.edu.
† E-mail: julia@cs.columbia.edu.
‡ E-mail: m.g.j.swerts@uvt.nl.

Submission received: 12 January 2005; revised submission received: 3 April 2006; accepted for publication: 4 May 2006

Given that current spoken dialogue systems are not sufficiently robust, there is need for sophisticated error-handling strategies to gracefully solve communication problems between the system and the user. Ideally, apart from strategies to *prevent* errors, error handling would consist of steps to immediately *detect* an error when it occurs and to interact with the user to *correct* the error in subsequent exchanges. To date, attempts to improve system performance have largely focused on improving ASR accuracy or simplifying the task, either by further constraining the domain and functionality of the system or by further restricting the vocabulary the system must recognize. Such studies include work on improved acoustic and semantic confidence scores (Ammicht, Potamianos, and Fosler-Lussier 2001; Andorno, Laface, and Gemello 2002; Bouwman, Sturm, and Boves 1999; Falavigna, Gretter, and Riccardi 2002; Guillevic, Gandrabur, and Normandin 2002; Moreno, Logan, and Raj 2001; Wang and Lin 2002; Zhang and Rudnicky 2001), on new system architectures for error handling (McTear et al. 2005; Prodanov and Drygajlo 2005; Torres et al. 2005), on new interfaces that are more user-friendly for error recovery (Bulyko et al. 2005; Karsenty and Botherel 2005; Sturm and Boves 2005), and on the use of error-recovery strategies that are based on analyses of human–human dialogues (Skantze 2005), including the use of facial expressions (Barkhuysen, Krahmer, and Swerts 2005).

However, as ASR accuracy improves, dialogue systems will be called upon to handle ever more complex tasks and ever less restricted vocabularies. So, it seems likely that spoken dialogue systems will, for the foreseeable future, always require effective error detection and repair strategies. In previous research (Hirschberg, Litman, and Swerts 1999, 2004), we identified new procedures to *detect* recognition errors, which perform well when tested on two different corpora, the TOOT and W99 corpora (train information and conference registration dialogues) collected using two different ASR systems (Sharp et al. 1997; Kamm et al. 1997). We found that prosodic features, in combination with information already available to the recognizer, such as acoustic confidence scores, grammar, and recognized string, can distinguish speaker turns that are misrecognized far better than traditional methods for ASR **rejection** (the system decision that its hypothesis is so weak that it should reprompt for fresh input), which use acoustic confidence scores alone. Related work has been done by Lendvai (2004) and Batliner et al. (2003). In the current study, we turn to the question of how people try to *correct* ASR errors in their interactions with machines and the role that prosody may play in identifying user corrections and in helping to analyze them.

Understanding how users attempt to correct system failures and why their attempts succeed or fail is important to improve the design of future spoken dialogue systems. For example, knowing whether they are more likely to repeat or rephrase their utterances, add new information or shorten their input, and how system behavior influences these choices can suggest appropriate on-line modifications to the system's interaction strategy or to the recognition procedure it employs. Determining which speaker behaviors are more successful in correcting system errors can also lead to improvements in the help information such systems provide. There is growing evidence that there is much variance in the way people react to system errors and that the variance can be explained on the basis of particular properties of the dialogue system or the dialogue context. In particular, dialogue confirmation strategies may hinder users' ability to correct system error. For instance, if a system wrongly presents information as being correct, as when it verifies information implicitly, users become confused about how to respond (Krahmer et al. 2001). Other studies have shown that speakers tend to switch to a prosodically "marked" speaking style after communication errors, comparing repetition corrections with the speech being repeated (Wade, Shriberg, and Price 1992; Oviatt et al. 1996;

Levow 1998; Bell and Gustafson 1999). Although this speaking style may be effective in problematic human–human communicative settings, there is evidence that suggests it leads to further errors in human–machine interactions (Levow 1998; Soltau and Waibel 2000). That corrections are difficult for ASR systems is generally explained by the fact that they tend to be *hyperarticulated*—higher, louder, longer—than other turns (Wade, Shriberg, and Price 1992; Oviatt et al. 1996; Levow 1998; Bell and Gustafson 1999; Shimojima, et al. 2001; Soltau and Waibel 1998, 2000; Soltau, Metze, and Waibel 2002), where ASR models are not well adapted to handle this special speaking style, although recent studies suggest that ASR systems are becoming less vulnerable to hyperarticulation (Bulyko et al. 2005).

So, repair strategies in human–machine interactions can be more or less effective. Therefore, increased knowledge about the efficiency of different correction strategies can lead to a number of possible courses of action. System strategy might be chosen to favor the type(s) of correction the system can most easily process. Or, having chosen a particular interaction strategy, the system repair strategy might be tuned to handle the correction types that that strategy is likely to produce. Alternatively, the system's dialogue manager might use the detection of corrections as a signal that it should modify its interaction strategy, either locally, by beginning a subdialogue for faster error recovery, or globally, by changing its initiative or confirmation strategies, or even directing the user to a human operator. Or, since corrections are often hyperarticulated, detection of a correction could serve as a signal to the ASR engine to run a recognizer trained on hyperarticulated speech in parallel with its normal processor, to better transcribe the speech. All of these possibilities, however, assume that user corrections can be detected by the system reliably during the dialogue.

In this article, we describe an analysis of user corrections of system error collected in the TOOT spoken dialogue system. In the next section, we describe the corpus itself and how it was collected and labeled. The corpus is suitable to gain insight into the different correction strategies that speakers exploit in different dialogue contexts and interaction styles. Then, we characterize the nature of corrections in this corpus in terms of when they occur, how well they are handled by the system, what distinguishes their prosody from other utterances, their relationship to the utterances they correct, and how they differ according to dialogue strategy. Then we present results of some machine-learning experiments designed to automatically distinguish corrections from other user input, using features that we derived as potentially useful from our descriptive analyses.

2. The Data

2.1 The TOOT Corpus

Our corpus consists of dialogues between human subjects and TOOT, a spoken dialogue system that allows access to train information from the Web via telephone. TOOT was collected to study variations in dialogue strategy and in user-adapted interaction (Litman and Pan 1999). It is implemented using an interactive voice response (IVR) platform developed at AT&T, combining ASR and text-to-speech with a phone interface (Kamm et al. 1997). The system's speech recognizer is a speaker-independent, hidden Markov model system with context-dependent phone models for telephone speech and constrained (rule-based) grammars defining vocabulary at any dialogue state. Whereas a "universal" grammar specifying all legal utterances was used at some points in the dialogue, seven smaller grammars were also used at many points in the dialogue (e.g., to recognize city names, days of the week, answers to yes/no questions,

etc.). Grammars were only written for originally expected answers; in other words, no specific grammar for corrections was built in.[1] Confidence scores for recognition were available only at the turn level and were based on acoustic likelihoods; thresholds for rejecting an utterance based on confidence scores were specified manually by the system designers and were set differently for different grammars. The platform supports barge-in. Subjects performed four tasks with one of several versions of the system that differed in terms of locus of initiative (system, user, or mixed), confirmation strategy (explicit, implicit, or none), and whether these conditions could be changed by the user during the task (adaptive vs. non-adaptive). In the adaptive version of the system, users were allowed to say *change strategy* at any point(s) in the dialogue. TOOT would then ask the user to specify new initiative and confirmation strategies, for example, *You are using the no confirmation strategy. Which confirmation strategy do you want to change to? No confirmation, implicit confirmation, or explicit confirmation?* TOOT's initiative strategy specifies who has control of the dialogue, whereas TOOT's confirmation strategy specifies how and whether TOOT lets the user know what it just understood. The fragments in Figure 1 provide some illustrations of how dialogues vary with strategy. For example, in user initiative mode, the system allows the user to specify any number of attributes in a single utterance. Thus, the system will let the user ignore specific questions. In the example in Figure 1, although the system asks for the day of the week, the user answers with the time, which can be recognized due to the use of the "universal grammar." In contrast, in both mixed and system initiative mode, when a specific question is asked, one of the restricted grammars is used to recognize the response. Finally, in universal and mixed but not system initiative mode, the system can ask both specific questions and open-ended questions (e.g., *How may I help you?*). Subjects were 39 students: 20 native speakers and 19 non-native, 16 women and 23 men. Dialogues were recorded and system and user behavior were logged automatically. The **concept accuracy** (CA) of each turn was manually labeled. If the ASR correctly captured all task-related information in the turn (e.g., time, departure, and arrival cities), the turn's CA score was 1 (*semantically correct*). Otherwise, the CA score reflected the percentage of correctly recognized task information in the turn. The dialogues were also transcribed and automatically scored in comparison to the ASR recognized string (the best hypothesis output by the ASR engine) to produce a **word error rate** (WER) for each turn. For the study described below, we examined 2,328 user turns (all user input between two system inputs) from 152 dialogues.

2.2 Labeling

To identify corrections in the corpus two authors independently labeled each turn as to whether or not it constituted a correction of a prior system failure (a rejection or CA error, which were the only system failure subjects were aware of) and subsequently decided upon a consensus label. Note that many of the discrepancies between labels were due to tiredness or incidental sloppiness of individual annotators, rather than true disagreement. Each turn labeled "correction" was further classified as belonging to one of the following categories: REP (repetition, including repetitions with differences in pronunciation or fluency), PAR (paraphrase), ADD (task-relevant content added), OMIT

1 Thus, if the system prompted for a single city, but the user also included a correction of a prior utterance (e.g., *No, at 10:30 p.m. I want to go to New York City*), the turn would be out of grammar. Coding our corpus for out of vocabulary turns and examining whether corrections are more likely to be out of grammar is an area for future work.

System Initiative, Explicit Confirmation
S: Which city do you want to go to?
U: Chicago.
S: Do you want to go to Chicago?
U: Yes.

User Initiative, No Confirmation
S: How may I help you?
U: I want to go to Chicago from Baltimore.
S: On which day of the week do you want to leave?
U: I want a train at 8:00.

Mixed Initiative, Implicit Confirmation
S: How may I help you?
U: I want to go to Chicago.
S: I heard you say go to Chicago. Which city do you want to leave from?
U: Baltimore.

Figure 1
Illustrations of various dialogue strategies in TOOT.

(content omitted), and ADD/OMIT (content both added and omitted). Repetitions were further divided into repetitions with pronunciation variation (PRON) (e.g., *yes* correcting *yeah*) and repetitions where the correction was pronounced using the same pronunciation as the original turn, but this distinction was difficult to make and turned out not to be useful. User turns that included both corrections and other speech acts were so distinguished by labeling them "2+." For example, the turn *I would like to go to Chicago from Baltimore change strategies system* contains not only an ADD correction, but also a request to adapt the system's dialogue strategies, followed by an inform of the desired initiative value. As another example, the turn *yes help* contains a REP correction, followed by a request for help. For user turns containing a correction plus one or more additional dialogue acts, only the correction is used for purposes of analysis below. We also labeled as "restarts" user corrections that followed non-initial system-initial prompts (e.g., *How may I help you?* or *What city do you want to go to?*); in such cases, system and user essentially started the dialogue over from the beginning.[2] Table 1 shows examples of each correction type and additional label for corrections of system failures on *I want to go to Boston on Sunday*. Note that the utterance on the last line of this figure is labeled 2+PAR, given that this turn consists of two speech acts: The goal of the no-part of this turn is to signal a problem, whereas the remainder of this turn serves to correct a prior error.

Each correction was also indexed with an identifier representing the closest prior turn it was correcting, so that we could investigate "chains" of corrections of a single failed turn by tracing back through subsequent corrections of that turn. Figure 2 shows a fragment of a TOOT dialogue with corrections labeled as discussed above.

3. Descriptive Analyses

This section presents the results of some descriptive analyses of the corrections we labeled in the TOOT corpus. We provide data on the distribution of different correction

2 Restarts occurred when either the user said the phrase *I'm done here* at any point in the dialogue, or answered *no* to the system's request to perform a database query (e.g., *Do you want me to find the trains from Baltimore to Chicago on Tuesday around 8:45 now?*).

Table 1
Example corrections of *I want to go to Boston on Sunday.*

Corr Type	Correction
REP	I want to go to Boston on Sunday
PAR	To Boston on Sunday
OMIT	I want to go to Boston
ADD	To Boston on Sunday at 8 p.m.
ADD/OMIT	I want to arrive Sunday at 8 p.m.
2+PAR	No, to Boston on Sunday

types, prosodic features of corrections, characteristics of correction chains, and variation in features of corrections as a function of dialogue strategy.

3.1 Correction Types

Of the correction types we labeled, the largest numbers were REPs and OMITs, as shown in Table 2, which shows overall distribution of correction types, and distributions for each type of system failure corrected, following either a misrecognized turn (with respect to concept accuracy) (Post-Misrec) or a rejected turn (Post-Rej) or correcting an earlier system failure (Non-Immed). (The last group includes corrections of earlier utterances that do not immediately follow a rejection or misrecognition.) Table 2 shows that 39% of TOOT corrections were simple repetitions of a previously rejected or misrecognized turn. Although this strategy is often suboptimal in correcting ASR errors (Levow 1998), REPs (45% WER error) and OMITs (52% error) were better recognized than ADDs (90% WER error) and PARs (72% WER error).

	Turn	Turn ID	Corr of	Corr Type
S:	*On which day of the week do you want to leave?*			
S:	*Sorry, I didn't hear anything from you. Please tell me your departure date.*			
U:	Um tomorrow	776	NA	NA
S:	*You can say the departure date. For example, you can say today or tomorrow. Or you can say Monday, Tuesday, Wednesday...*			
U:	Tomorrow	777	776	REP
S:	*You can say the departure date. For example, you can say today or tomorrow. Or you can say Monday, Tuesday, Wednesday, Thursday, Friday...*			
U:	Saturday	778	777	PAR
S:	*Do you want to leave on Saturday?*			
U:	Yes	779	NA	NA
S:	*What time do you want to leave?*			
U:	Eight o'clock a.m.	780	NA	NA
S:	*Do you want to leave around 10:00 p.m.?*			
U:	Eight o'clock	781	780	OMIT
S:	*Do you want to leave around 10:00 p.m.?*			
U:	No	782	NA	NA
S:	*What time do you want to leave?*			
U:	Eight o'clock a.m.	783	781	ADD

Figure 2
Toot dialogue fragment with correction labels.

Table 2
Distribution of correction types.

Type	ADD		ADD/OMIT		OMIT		PAR		REP		*N*
Total	51	8%	14	2%	215	32%	127	19%	265	39%	672
Post-Misrec	39	7%	13	3%	203	40%	90	18%	173	32%	518
Post-Rej	8	6%	0	0%	9	7%	36	28%	75	59%	128
Non-Immed	4	15%	1	4%	3	12%	1	4%	17	65%	26

There was no significant difference either in the number of corrections produced ($\chi = 2.44, p = .12$) or in correction type ($\chi^2 = 5.07, p = .28$) between our native speaker subjects and non-native speakers. However, what the user was correcting did influence the type of correction chosen. Table 2 shows that corrections of misrecognitions (Post-Misrec) were more likely to omit information present in the original turn (OMITs), whereas corrections of rejections (Post-Rej) were more likely to be simple repetitions. The latter finding is not surprising because the rejection message for tasks was always a close paraphrase of *Sorry, I can't understand you. Can you please repeat your utterance?* However, it does suggest the surprising power of system directions and how important it is to craft prompts to favor the type of correction most easily recognized by the system.

3.2 Prosodic Features of Corrections

In part to test the hypothesis that corrections tend to be hyperarticulated (slower and louder speech that contains wider pitch excursions and more internal silence), we examined the following features for each user turn: maximum and mean fundamental frequency values (f0 Max, f0 Mean); maximum and mean energy values (RMS Max, RMS Mean); total duration; length of pause preceding the turn (Prior Pause); speaking rate (Tempo), calculated in syllables per second (sps); and amount of silence within the turn (% Silence).[3] f0 and RMS values, representing measures of pitch excursion and loudness, were calculated from the output of Entropic Research Laboratory's pitch tracker, *get_f0* (Talkin 1995), with no postcorrection. Timing variation was represented by four features: Duration of turn and length of pause between turns was hand labeled. Speaking rate was approximated in terms of syllables in the recognized string per second. % Silence was defined as the percentage of zero frames in the turn, calculated from the pitch track; this feature approximates the percentage of time within the turn that the speaker was silent.

To ensure that our results were speaker independent, we calculated mean values for each speaker's corrections and non-corrections for every feature. Then, for each feature, we created vectors of speaker means for correction and non-correction turns and performed paired *t* tests on the paired vectors. For example, for the feature "f0 Max," we calculated mean maxima for correction turns and for non-corrections for each of our thirty-nine speakers. We then performed a paired *t* test on these

3 Although the features were automatically computed, turn beginnings and endings were hand segmented in dialogue-level speech files, as the turn-level files created by TOOT were not available. Because of some system/user overlap in the recordings, we were able to calculate prosodic features for only 1,975 user turns.

thirty-nine pairs of means to derive speaker-independent results for differences in f0 maxima between corrections and non-corrections. Note, however, that there were overall differences in the corrections produced by native and non-native speakers, normalized by value of first turn in task: mean f0 was higher for native speakers than for non-native speakers (t stat $= -2.72, df = 602, p = .0067$), tempo was faster ($t$ stat $= -3.18, df = 670, p = .0015$), and duration was shorter (t stat $= 2.20, df = 670, p = .028$). These differences do not occur in non-correction utterances.

Our results provide some explanation for why corrections are more poorly recognized than non-corrections because they indicate that corrections are indeed characterized by prosodic features associated with hyperarticulation. Table 3 shows that corrections differ from other turns in that they are longer, louder, higher in pitch excursion, follow longer pauses, and contain less internal silence than non-corrections. All but the latter difference supports the hypothesis that corrections tend to be hyperarticulated.

To confirm this hypothesis, two of the authors labeled each turn in the corpus for evidence of perceptual hyperarticulation, following (Wade, Shriberg, and Price 1992). Fifty-two percent of corrections in the corpus have some perceptual hyperarticulation, compared with only 12% of other turns. Too, hyperarticulated corrections are more likely to be misrecognized than other corrections (70% misrecognitions vs. 52%). However, it is important to note that only 59% of misrecognized corrections in the corpus are also hyperarticulated, so recognition failure for a considerable portion of corrections must be explained in some other way. There is still a large number of misrecognized corrections that show no perceptual evidence of hyperarticulation.

In our earlier analysis of prosodic differences between correct and incorrectly recognized turns (Hirschberg, Litman, and Swerts 2004), we also found that misrecognized turns differed from correctly recognized turns in f0, loudness, duration, and timing—all features associated with hyperarticulation. In addition, more misrecognitions are hyperarticulated than are correctly recognized turns. But when we excluded perceptually hyperarticulated turns from our prosodic analysis, we found that misrecognized turns were still prosodically different from correctly recognized turns, in the same ways. We hypothesized there that misrecognitions might exhibit tendencies toward hyperarticulation that are imperceptible to human listeners, but not to ASR engines. The same may also be true of non-hyperarticulated, but still prosodically distinct corrections. When we exclude hyperarticulated utterances from our corpus

Table 3
Corrections versus non-corrections by prosodic feature.

Feature	t stat	Mean corr - non-corr	p
f0 Max*	3.79	17.76 Hz	$< .001$
f0 Mean	0.23	−4.12 Hz	.823
RMS Max*	4.88	347.75	$< .001$
RMS Mean*	2.57	63.44	.014
Duration*	6.68	1.16 sec	$< .001$
Prior pause*	2.17	0.186 sec	.036
Tempo	1.78	−0.15 sps	.246
% Silence*	4.75	−0.05%	$< .001$

*Significant at a 95% confidence level ($p \leq .05$)

and reanalyze prosodic features of corrections versus non-corrections, we find significant differences in duration, rms maximum, rms mean, tempo, and amount of turn-internal silence as we did with the corpus as a whole. So, again, even when corrections are not perceptibly hyperarticulated, they share some acoustic tendencies with turns that are.

3.3 Correction Chains

As noted above, corrections in the TOOT corpus often take the form of chains of corrections of a single original error. Looking back at Figure 2, for example, we see two chains of corrections: In the first, which begins with the misrecognition of turn 776 (*Um, tomorrow*), the user repeats the original phrase and then provides a paraphrase (*Saturday*), which is correctly recognized. In the second, beginning with turn 780, the time of departure is misrecognized. The user omits some information (*a.m.*) in turn 781, but without success; an ADD correction follows, with the previously omitted information restored, in turn 783.

Distance of a correction from the original misrecognized turn—whether calculated as position in chain (e.g., *Saturday* in Figure 2 is the second in the chain correcting turn 776) or further in number of turns from that original error (e.g., *Saturday* here is also two turns from the original error)—correlates significantly with prosodic variation. An analysis of the relationship between both distance measures and our prosodic features (using Pearson's product–moment correlation) shows significant correlations of distance in chain or from original error with f0 maximum ($r = .20$, $p < .001$; $r = .21$, $p < .001$) and mean ($r = .27$, $p < .001$; $r = .29$, $p < .001$), rms maximum ($r = -.09$, $p < .02$; $r = -.12$, $p < .005$) and mean ($r = -.12$, $p < .0025$; $r = -.16$, $p < .001$), absolute duration ($r = .14$, $p < 0$; $r = .16$, $p < .001$) and duration in number of words ($r = .11$, $p < .01$; $r = .12$, $p < .005$), length of preceding pause ($r = .11$, $p < .005$; $r = .10$, $p < .01$), and speaking rate ($r = -.05$, $p < .01$; $r = -.10$, $p < .02$). The more distant a correction is, in short, the higher it is in pitch, the softer it is, the longer it is, the greater is its preceding pause, and the more slowly it is spoken. In addition, more distant corrections are also more likely to be misrecognized; for distance in turns there is a (negative) significant correlation for concept accuracy ($r = -.13$, $p < .001$), whereas both word and concept accuracy decline significantly by position in chain ($r = -.08$, $p < .05$; $r = -.15$, $p < .001$). Table 4 shows the mean concept accuracy of corrections for chain position through 8 (higher numbers are very small) in the corpus. So, as speakers must try again and again to correct an error, their attempts appear to become ever less likely to succeed, perhaps because their prosodic behavior changes in ways that do not help the recognition process. Curiously, however, our perceptual measure of hyperarticulation is not significantly correlated with either of these distance measures. So, although speakers modify their speech in ways generally consistent with hyperarticulation, their corrections do not necessarily become more hyperarticulated as their attempts to correct continue. Another curious

Table 4
Mean concept accuracy by correction position in Chain.

Position	1	2	3	4	5	6	7	8
N	311	143	84	49	25	15	10	4
Error	.43	.57	.63	.51	.60	.87	.70	1.00

finding is that corrections that are more distant from the turn they immediately correct (e.g., in Figure 2, turn 783 is more distant from the turn it corrects (781) than turn 781 is from the turn *it* corrects, which is 780) tend to be *more* accurately recognized than turns that are closer. Yet, prosodically, these turns are very like distant turns in a chain or from the original error, being higher in f0 maximum and mean, lower in rms maximum and mean, and longer in seconds and number of words. So, in the one case these prosodic changes might be thought to lead to recognition error, where in the other they occur with better recognized corrections.

3.4 Variation by Dialogue Strategy

Dialogue strategy clearly affects the type of correction users make and whether it is successful or not. For example, users more frequently repeat their misrecognized utterance in the SystemExplicit (75% of corrections are repetitions) condition than in the MixedImplicit or UserNoConfirm (both 37% REP); the latter conditions have larger proportions of OMITs and paraphrases. Perhaps this disparity is partly explained by the larger proportion of corrections that follow rejections in the SystemExplicit condition (39% vs. 22% and 19%). Overall, SystemExplicit turns are rejected 6% of the time, whereas the other conditions have about 10% rejections. Table 5 shows differences in mean length of tasks, number of corrections, number of misrecognitions, and number of misrecognized corrections by dialogue strategy. Again, misrecognitions were defined in terms of concept accuracy (turns with CA < 1); misrecognized corrections refer to the intersection of user terms that were coded as both corrections and misrecognitions. The fewer misrecognitions, corrections, and misrecognized corrections per task in the SystemExplicit condition may well explain user ratings of the various systems (non-adapt) in the original experiments (Litman and Pan 1999): When asked to say whether they would be likely to use such a system in the future, on a 1–5 scale, subjects scored SystemExplicit 3.5, MixedImplicit 2.6, and UserNoConfirm 1.7. User satisfaction scores were similar: Where 40 is the highest score, users gave SystemExplicit 31.25, MixedImplicit 24, and UserNoConfirm 22.1. So, SystemExplicit is preferred by users, even though MixedImplicit on average takes fewer turns to accomplish a task, suggesting that the large number of misrecognitions and consequent need for correction has a large impact on user preferences. This is consistent with performance functions derived from evaluations of TOOT (Litman and Pan 1999).

Perhaps because correction chains often end unsuccessfully, users frequently "restart" a task within a session. Most restarts occurred in the MixedImplicit and UserNoConfirm conditions and were rarely successful. In non-adaptive tasks, 42% of corrections in the MixedImplicit condition were restarts and 31% in the UserNoConfirm,

Table 5
Corrections by system strategy.

Means per task	SystemExplicit	MixedImplicit	UserNoConfirm
# Turns	13.4	11.7	16.2
# Corrs	1.3	4.6	7.1
# Misrecs	2.8	6.4	9.4
# Misrec'd Corrs	0.3	3.2	4.8

whereas none occurred in the SystemExplicit condition. Restarts were misrecognized 77% of the time, compared to 65% of first turns in task. They thus seem to have been a worse strategy than initiating a new task and might prove a useful diagnostic for changing system strategy—or summoning a human operator.

4. Predicting Corrections

The previous section showed that corrections differ significantly from non-corrections prosodically, being higher in pitch, louder, longer, with longer pauses preceding them and less internal silence. In addition, they are misrecognized more frequently than non-corrections—although they are no more likely to be rejected by the system. And corrections more distant from the error they correct tend to exhibit greater prosodic differences and are recognized more poorly, suggesting that users are not learning to modify their own behavior to improve system performance. So, dealing with corrections is a particularly difficult task for both users and systems. We also found that system dialogue strategy—the amount of initiative users are allowed to exercise in controlling the flow of the dialogue and the type of confirmation strategy the system adopts—affects users' choice of correction type (e.g., directly repeating vs. paraphrasing misrecognized information). In the following, we turn to the question of identifying user corrections automatically, from prosodic features as well as other features that are readily available to a spoken dialogue system. In Section 4.1, we describe the features we use for our machine-learning experiments. Section 4.2 presents the results of those experiments. Section 4.3 presents further experiments using additional classifications and features, motivated by our descriptive results. In the final section, we summarize our conclusions and describe future research directions.

4.1 Features

In this section we describe the features used in the machine-learning experiments and the motivation behind their selection. The entire feature set is presented in Figure 3 and includes only features that could be automatically available to a dialogue system.

4.1.1 Prosodic Features. Above we showed that corrections were significantly longer, louder, higher in pitch excursion, and followed longer pauses than other turns. Thus, we expected that these features would be useful in identifying corrections automatically. We examined maximum and mean fundamental frequency values (**f0max**, **f0mn**) as indicators of pitch range; maximum and mean energy values (**rmsmax**, **rmsmn**) as indicators of loudness; total duration of the speaker turn (**dur**); length of pause preceding the turn (**ppau**); speaking rate (**tempo**); and amount of silence within the turn (**zeros**). The features were measured as indicated above. Table 6 shows the overall means and standard deviations for these features over the corpus.

4.1.2 ASR Features. Since corrections in our corpus were misrecognized more frequently than non-corrections (Swerts, Litman, and Hirschberg 2000), we included a set of ASR features that were derived from TOOT's speech recognition component and its outputs: the grammar used as the ASR language model at each dialogue state (**gram**), the string recognized by the ASR system as its best hypothesis (**str**), and the turn-level acoustic

Prosodic (PROS) :

Raw (raw values): f0max, f0mn, rmsmax, rmsmn, dur, ppau, tempo, zeros
Norm1 (values normalized by first turn in dialogue): f0max1, f0mn1, rmsmax1, rmsmn1, dur1, ppau1, tempo1, zeros1
Norm2 (values normalized by previous turn in dialogue): f0max2, f0mn2, rmsmax2, rmsmn2, dur2, ppau2, tempo2, zeros2

ASR (ASR) : gram, str, conf, ynstr, nofeat, canc, help, wordsstr, syls, rejbool

System Experimental (SYS) : inittype, conftype, adapt, realstrat

Dialogue Position (POS) : diadist

Dialogue History (DIA) :

PreTurn : value of PROS and ASR features for preceding turn (e.g., pref0max)
PrepreTurn : value of PROS and ASR features for turn preceding preceding turn (e.g., ppref0max)
Prior : for each Boolean-valued feature (ynstr, nofeat, canc, help, rejbool), the number/percentage of prior turns exhibiting the feature (e.g., priorynstr-num/priorynstrpct)
PMean : for each continuous-valued PROS and ASR feature, the mean of the feature's value over all prior turns (e.g., pmnf0max)

Figure 3
Feature set for predicting corrections.

Table 6
Means and standard deviations for prosodic features over all turns.

	f0max (Hz)	**f0mn** (Hz)	**rmsmax**	**rmsmn**	**dur** (sec)	**ppau** (sec)	**tempo** (sps)	**zeros** (%)
Mean	227	163	1612	396	1.92	.71	2.48	44
S.D.	77	44	1020	261	2.44	.79	1.37	17

confidence score it produced (**conf**).[4] As subcases of the **str** feature, we included Boolean features representing whether or not the recognized string included the strings *yes* or *no* (**ynstr**), some variant of *no*, such as *nope* (**nofeat**), *cancel* (**canc**), or *help* (**help**), as these lexical items often occurred during problem resolution. To estimate durational features, we approximated the length of the user turn in words (**wordsstr**) and in syllables (**syls**) from the **str** feature, and we added a Boolean feature identifying whether or not the turn had been rejected by the system (**rejbool**).

4.1.3 System Experimental Features. Our descriptive study showed that differences in dialogue strategy affect the type and success of user corrections. For example, TOOT users more frequently repeat their misrecognized turns and produce the fewest corrections per task when TOOT has the initiative and explicitly confirms all user input. So, we hypothesized that system conditions might prove important in our learning experiments. We thus include features representing the system's current initiative and confirmation strategies (**inittype**, **conftype**), whether users could adapt the system's dialogue strategies (**adapt**), and the combined initiative and confirmation setting (**realstrat**).

4 Confidence scores ranged from −0.087662 to −9.884418.

4.1.4 Dialogue Position and History Features. We also showed that the further a correction is from the original error, the less likely it is to be recognized correctly, and the stronger the correlation with prosodic deviation from the mean values over a speaker's turns (e.g., more distant corrections are higher in pitch than closer corrections). As a first approximation of this distance feature, we included the feature **diadist**—distance of the current turn from the beginning of the dialogue.

In addition, previous research (Litman, Walker, and Kearns 1999; Walker et al. 2000) has shown that features of the dialogue as a whole and features of more local context can be helpful in predicting "problematic" dialogues. So we looked at a set of features summarizing aspects of the prior dialogue for both the absolute number of times prior turns exhibited certain characteristics (e.g., contained a key word like *cancel*—**priorcancnum**) and the percentage of the prior dialogue containing one of these features (e.g., **priorcancpct**). We also examined means for all our continuous-valued features over the entire dialogue preceding the turn to be predicted (**pmn**__), such as **pmnsyls**, the mean length of prior turns calculated in number of syllables per turn. Finally, we examined more local contexts, including all features of the preceding turn (**pre**__) and for the turn preceding that (**ppre**__).

It seemed particularly likely that lexical features of the local context—such as whether a user had asked for help recently, or tried to cancel out of an exchange, or replied *no* to a system query—might prove useful in identifying corrections.[5] Also, whether a prior turn had been rejected was clearly a useful cue to the identification of the current turn as a correction, since users generally supplied a correction when explicitly asked.

4.2 Machine-learning Experiments

In this section we investigate whether the features described in Section 4.1 (or interesting subsets of them) can in fact be used to accurately *predict* whether a turn will be a correction or not. We describe experiments using the machine-learning program RIPPER (Cohen 1996) to automatically induce such prediction models. RIPPER takes as input the classes to be learned, the names and possible values of a set of features, and training data specifying the class and feature values for each training example. For our experiments, the features presented in Figure 3 comprise the independent variables for our learning experiments. The dependent variable to be learned, **correction** (T) versus **non-correction** (F), corresponds to the hand-labeled observations described above. Given a vector of values for the independent and dependent variables for each speaker turn, RIPPER outputs a classification model for classifying future examples. The model is learned using greedy search guided by an information gain metric and is expressed as an ordered set of *if-then* rules. When multiple rules are applicable, RIPPER uses the first rule it finds. When no rules are applicable, RIPPER classifies the turn as a non-correction (F) by default.

Table 7 shows the performance of the learned classification models for some of the feature sets we examined; all performance figures are estimated using 25-fold cross-validation on the 2,328 turns in our corpus. The Features column identifies the set of features (as defined in Figure 3) used to learn the model. The second column, DIA, indicates which type of dialogue history features (PreTurn, PrepreTurn, Prior,

5 Recall that these are lexical features from the recognized string, not from the actual user transcript.

Table 7
Estimated error, recall, precision, and $F_\beta = 1$ for predicting corrections.

			class = T			class = F		
Features	DIA	Error ± SE	Rec.	Prec.	$F_\beta = 1$	Rec.	Prec.	$F_\beta = 1$
Raw+ASR+SYS+POS	PreTurn	15.72 ± 0.80	70.61	74.96	.72	89.95	88.28	.89
Raw+ASR+SYS+POS	all	16.16 ± 0.58	69.80	74.65	.72	90.12	87.82	.89
PROS+ASR+SYS+POS	all	16.38 ± 0.61	69.01	74.05	.71	89.60	87.61	.88
ASR	all	16.41 ± 0.93	69.93	72.39	.70	88.76	87.7	.88
ASR+SYS+POS	all	17.01 ± 0.78	73.73	73.38	.73	88.68	89.00	.89
ASR+SYS+POS	none	18.60 ± 0.81	56.48	72.79	.63	91.33	83.76	.87
Raw+ASR+SYS+POS	none	18.68 ± 0.67	58.45	71.64	.64	90.37	84.17	.87
ASR+PROS	none	19.29 ± 0.78	54.54	69.97	.61	90.25	82.90	.86
POS+PROS	none	19.59 ± 0.73	52.96	69.70	.60	90.38	82.47	.86
Raw	all	19.68 ± 0.78	55.62	70.89	.62	90.64	83.33	.87
PROS	all	20.33 ± 0.90	56.45	69.23	.61	89.43	83.42	.86
ASR+POS	none	20.40 ± 0.79	52.20	71.99	.60	91.43	82.41	.87
PROS	none	20.53 ± 0.81	54.86	71.72	.62	90.78	83.07	.87
conf+rejbool	all	21.23 ± 0.93	59.70	65.97	.62	87.05	84.05	.85
ASR+SYS	none	23.46 ± 0.72	51.55	63.40	.56	87.53	81.65	.84
ASR	none	24.19 ± 0.84	45.93	60.99	.52	87.80	79.90	.84
Raw	none	25.35 ± 0.93	42.26	59.46	.48	88.29	78.97	.83
POS	none	29.00 ± 1.02	0.00	–	–	99.94	70.99	.83
SYS	none	29.00 ± 1.02	0.00	–	–	100.00	71.00	.83

Prerejbool baseline error = 25.70; majority baseline error = 28.99

and/or PMean) were also included in the feature set; these features represent the same types of information (e.g., **f0max**) that the Features column denotes, but for one or more *previous* turns in the dialogue. The third column shows the mean error and standard error (SE) predicted for the model specified by the first two columns. When error estimates in different rows differ by plus or minus twice the standard error, they are significantly different (Cohen 1995). The remaining columns show the mean *recall*, *precision*, and $F_\beta = 1$ for corrections (focus class = T) and non-corrections (focus class = F), respectively.[6] For comparison purposes, we compare our predictions to two potential baselines. The Majority baseline predicts that all turns are non-corrections (the majority class of F), and has a classification error of 28.99%. The Prerejbool baseline predicts that all turns following rejected turns (**prerejbool** = T) are corrections—since after rejections, TOOT asks users to repeat their turn[7]—and all others are non-corrections; this baseline gives a classification error of 25.70%.

The first question addressed in our experiments is whether or not corrections can be predicted significantly better than our baselines. Table 7 shows that in fact they can. Our best-performing feature set (Raw+ASR+SYS+POS, DIA = PreTurn) cuts the majority baseline error almost in half, from 28.99% to 15.72%, and predicts significantly better than the rejection-based baseline as well. This feature set includes raw versions of all our prosodic features and all of the non-prosodic features, for both the turn being classified

6 *Recall* is the percentage of actual members of a class that are identified, whereas *precision* is the percentage of predicted class members that are in fact members. The definition of F_β is $\frac{(\beta^2+1)\text{PrecisionRecall}}{\beta^2\text{Precicison}+\text{Recall}}$; $\beta = 1$ equally weights precision and recall. These values are computed using our own cross-validation program, while error is computed using RIPPER's cross-validation option.

7 Although users are asked to repeat their turn, 29% of the turns after rejections are *not* in fact corrections (e.g., the user instead asks for help or asks the system to repeat the prompt).

and the immediately prior turn. Note that even if **all** of the available features are used for learning (i.e., the normalized versions of prosodic features and all of the various history features (PROS+ASR+SYS+POS, DIA = all, error = 16.38%)), performance is statistically comparable to this model.[8] In addition, the recall, precision and $F_{\beta} = 1$ values in Table 7 show that corrections are generally predicted with better precision than recall whereas the reverse holds for non-corrections, and that non-corrections (the majority class) are easier to accurately predict than corrections.

We next turn to an examination of the contribution of the different types of features we used for prediction. First, we consider the utility of our non-prosodic features. Table 7 shows that, using only non-prosodic features (ASR, SYS, POS), corrections can still be predicted with an accuracy statistically equivalent to our best results. That is, using all feature types (PROS+ASR+SYS+POS, DIA = all, error = 16.38%) is equivalent to using only non-prosodic features (ASR+SYS+POS, DIA = all, error = 17.01%). Similarly, restricting our feature set to the ASR-derived subset of our non-prosodic features (ASR, DIA = all, error = 16.41%) or removing all dialogue history (ASR+SYS+POS, DIA = none, error = 18.60%) yields results equivalent to our best-performing classifier. However, when only those ASR features derived from the acoustic confidence score (i.e., **conf**, **preconf**, **ppreconf**, **pmnconf**, **rejbool**, **prerejbool**, **pprerejbool**, **priorrejboolnum**, **priorrejboolpct**) are used for prediction, then performance does significantly degrade (conf+rejbool, DIA = all, error = 21.23%). So, it appears that there are numerous ways to classify corrections successfully, using various combinations of feature types. This finding is an important one because it suggests that systems that have access to restricted kinds of information can still hope to identify user corrections with some confidence. In particular, simply using information available to current ASR systems, such as acoustic confidence score, recognized string, grammar, and features derived from these, produces classification results equivalent to our best-performing classifier. A caveat here is that some of the features in this ASR feature set (e.g., grammar and recognized string) are less likely to generalize from task to task.

Turning now to the role of prosodic features in classifying corrections, Table 7 shows that use of only non-prosodic features (ASR+SYS+POS, DIA = all, error = 17.01%)[9] slightly (but not quite significantly) outperforms use of only raw prosodic features (Raw, DIA = all, error = 19.68%). However, using raw prosodic features alone (error = 19.68%) is comparable to using only ASR features alone (ASR, DIA = all, error = 16.41%). And both significantly outperform the majority class and rejection-based baselines. Note also that prediction from raw prosodic features alone (19.68%) is not improved by the inclusion of their normalized versions (PROS, DIA = all, error = 20.33%). Thus, ASR-derived features and prosodic features seem to provide equally successful classifications of user corrections. Since ASR-derived features, in particular, acoustic confidence score, are currently used by spoken dialogue systems to determine when to *reject* a turn, our results suggest that such features can also be useful for identifying corrections. Although prosodic features are rarely made use of in spoken dialogue systems, they would, in fact, seem more likely to generalize across tasks and recognizers than the ASR features.

Now we turn to the issue of how useful features of the dialogue history are in classifying corrections. Recall that our best-performing ruleset used only a limited dia-

8 Note that removing features sometimes changes performance, which might indicate a weakness in RIPPER's feature selection process.

9 Recall that DIA = all includes only the same type of features as for the current utterance, in this case only non-prosodic history features.

if (dur ≥ 3.89046) ∧ (preconf ≤ –0.645234) ∧ (zeros ≤ 0.539474) **then** *T* (153/10)
if (dur ≥ 0.851477) ∧ (preconf ≤ –2.20989) ∧ (zeros ≤ 0.442509) **then** *T* (114/47)
if (syls ≥ 3) ∧ (preppau ≥ 0.393313) ∧ (gram = universal) ∧ (pretempo ≤ 2.30808) **then** *T* (52/16)
if (preconf ≤ –3.85311) ∧ (predur ≤ 0.982059) ∧ (prerejbool = T) **then** *T* (51/12)
if (dur ≥ 0.736544) ∧ (diadist ≥ 9) ∧ (syls ≥ 4) ∧ (conftype = Implicit) **then** *T* (32/10)
if (prestr contains help) ∧ (preppau ≤ 1.35977) **then** *T* (46/13)
if (syls ≥ 2) ∧ (preppau ≥ 0.509916) ∧ (pref0mn ≤ 118.773) **then** *T* (35/22)
if (dur ≥ 0.66384) ∧ (predur ≥ 0.698772) ∧ (conf ≤ –3.16533) ∧ (syls ≥ 4) **then** *T* (24/11)
if (pretempo ≤ 0.437603) ∧ (preconf ≥ –0.393746) **then** *T* (15/2)
if (pretempo ≤1.39342) ∧ (preconf ≤ –4.06433) ∧ (prewordsstr ≤ 3) **then** *T* (22/15)
else F (1495/131)

Figure 4
Best performing ruleset (Raw+ASR+SYS+POS, DIA = PreTurn).

logue history—features from the preceding turn (Raw+ASR+SYS+POS, DIA = PreTurn, error = 15.72%). While adding features of the turn two turns back (PrepreTurn__) and of the dialogue as a whole (Prior__ and PMean__) does not significantly change the error (Raw+ASR+SYS+POS, DIA = all, error = 16.16%), removing the features of the immediately previous turn from the dialogue history does in fact cause a significant increase in error rate (Raw+ASR+SYS+POS, DIA = none, error = 18.68%). However, as discussed above, when only non-prosodic features are considered (ASR+SYS+POS), there is no significant difference between DIA = all and DIA = none. So, it seems that features of the immediate local context can improve our ability to classify corrections accurately when prosodic features are included, but adding a larger local context window and a global context does not improve over these results. Contextual features seem particularly important to performance when only raw prosodic features are considered (Raw, DIA = all, error = 19.68%). When the raw prosodic features of the dialogue history are removed, the error rate dramatically increases (Raw, DIA = none, error = 25.35%). However, if the normalized prosodic features (which themselves encode much of the historical information) are also included, then removing the DIA versions of these features does not significantly degrade performance (PROS, DIA = all, error = 20.33% vs. PROS, DIA = none, error = 20.53%). We might explain the larger role that prosodic context plays in classification by returning to the differences we found between prosodic features of corrections and non-corrections, described in Section 3. In our descriptive analyses we found that prosodic features such as pitch, duration, and loudness reliably distinguish corrections based on relative differences between the two types of turns, not absolute differences. In prediction also, it seems that some form of normalization by context improves the performance of prosodic features.

When we examine which class of features performs best in the absence of contextual information, we see that the prosodic features (PROS, DIA = none, error = 20.53%) significantly outperform the ASR-derived features (ASR, DIA = none, error = 24.19%), which in turn significantly outperform either of the remaining feature types (POS and SYS). Table 7 also shows the cases in which the addition of new sources of knowledge improves prediction performance. For DIA = none, the statistically significant improvements involve adding the feature **diadist** (distance of the current turn from the beginning of the dialogue): For example, ASR+POS (error = 20.4%) outperforms both ASR (error = 24.19%) and POS (error = 29%), and ASR+SYS+POS (error = 18.6%) outperforms ASR+SYS (error = 23.46%). Again, these are features that are easily made available to current spoken dialogue systems.

The classification model learned from the best-performing feature set in Table 7 is shown in Figure 4. Rules are presented in order of importance in classifying data. The

first rule RIPPER finds with this feature set specifies that if the duration of the current turn is $\geq$ 3.89046 seconds, and if the acoustic confidence score of the prior turn is ≤ -0.645234, and if the percentage of silence in the current turn is $\leq$ 53.9474%, then predict that the turn is a correction; this rule correctly predicts 153 corrections and incorrectly predicts that 10 non-corrections are corrections. So, this rule applies when the previous turn has a low confidence score and the current turn exhibits some marked prosodic features. The fourth rule predicts a correction after a previous rejection, but only when the rejected turn was relatively short with a low confidence score. The fifth rule predicts a correction when TOOT uses a particular confirmation strategy for turns that are relatively long and far from the beginning of the dialogue. The sixth rule predicts a correction when the previous turn is spoken soon after the prompt, and contains the problem indicator *help*. Note that this use of the domain-independent *help* is the only reference to a lexical item in this ruleset. This ruleset includes features from all of the feature subsets in our inventory (PROS, ASR, SYS, POS, DIA). For the current turn, the feature types that appear in the rules are PROS (**dur**, **zeros**), ASR (**conf**, **gram**, **syls**), SYS (**conftype**), and POS (**diadist**). Of the previous turn's features, only two feature sets emerge as important: PROS (**pref0mn**, **predur**, **preppau**, **pretempo**) and ASR (**preconf**, **prestr**, **prewordstr**, **prerejbool**). Furthermore, within a feature set such as PROS, the useful features of the current and previous turns differ somewhat (e.g., **zeros** is useful for the current turn, whereas **tempo** is useful for the prior turn), suggesting important differences in the prosodic characteristics of corrections versus the turns they follow.

When we look at a ruleset produced using only features commonly available to current dialogue systems, such as ASR+SYS+POS (DIA = all), we see that creative use of these features could in fact support correction classification (Figure 5). For example, the fourth rule predicts that the current turn is a correction when it is not too short, and when the **pre**__ turn indicates awareness (evidenced by the presence of *no*) of a problem in the **ppre**__ turn (which was recognized with low confidence). This ruleset uses both ASR (**gram**, **nofeat**, **syls**) and SYS (**conftype**) features of the current turn; although only one rule in fact makes use of SYS features. For the contextual DIA features, only the ASR features occur in the rule-set: PreTurn (**preconf**, **prestr**, **prenofeat**, **prerejbool**), PrepreTurn (**ppreconf**, **ppreynstr**), and Prior and PMean (**pmnconf**, **priorynstrpct**, **pmnwordsstr**, **priorrejnum**). Comparing this ruleset to the previous one (Figure 4), we see that where timing features (**dur**, **predur**, **zeros**, **pretempo**, **preppau**) appear often when prosodic features are available, related features such as **syls** and **wordsstr** (from which, e.g., **tempo** is estimated) may be compensating in this ruleset. And of course the rejection feature (**prerejbool**) itself is a function of the confidence score of the prior turn. Note also that lexical features of the recognized string (**nofeat**, **prenofeat**, **ppreynstr**,

if (pmnconf ≤ -2.67657) $\wedge$ (syls ≥ 3) $\wedge$ (gram = universal) **then** *T* (287/70)
if (preconf ≤ -3.0156) $\wedge$ (prerejbool = T) $\wedge$ (nofeat = T) **then** *T* (26/5)
if (preconf ≤ -4.0034) $\wedge$ (ppreynstr = F) $\wedge$ (prerejbool = T) **then** *T* (42/16)
if (ppreconf ≤ -2.29048) $\wedge$ (syls ≥ 3) $\wedge$ (prenofeat = T) **then** *T* (31/2)
if (prestr contains help) **then** *T* (55/27)
if (syls ≥ 3) $\wedge$ (pmnwordsstr ≥ 2.05714) $\wedge$ (conftype = Implicit) $\wedge$ (priorrejnum ≥ 1) **then** *T* (38/11)
if (preconf ≤ -3.94692) $\wedge$ (syls ≥ 3) $\wedge$ (priorynstrpct ≤ 0.142857) $\wedge$ (pmnwordsstr ≥ 1.66667) **then** *T* *T* (17/2)
else F (1520/179)

Figure 5
Ruleset for non-prosodic features (ASR+SYS+POS, DIA = all).

prestr, **priorynstrpct**) emerge as quite useful in this ruleset—especially as contextual features. So, what the system has recognized in prior turns is a good predictor of whether the current turn is a correction. Also note that the overall verbosity of the previous dialogue (**pmnwordsstr**) appears in two of the rules.

An example of a ruleset learned from only prosodic features (Raw, DIA = all, from Table 7) is shown in Figure 6. This ruleset is notably terser than those shown in Figures 4 and 5 and includes primarily timing-based features (current turn features **dur**, **zeros**, and **tempo**; local contextual feature **pretempo**; and dialogue-level features **pmndur** and **pmnppau**). However, all prosodic feature types but f0 appear at least once in the ruleset, and features specific to the current turn differ from those relevant to different types of dialogue history. As with our previous descriptive findings discussed in Section 3, this ruleset shows that corrections are longer, louder, follow longer pauses, and contain less internal silence than non-corrections, and that these features can be used successfully to identify them.

4.3 Other Experiments

The machine-learning experiments described in Section 4.2 were motivated by our long-term goal to incorporate a correction predictor into future versions of our spoken dialogue system. As such, the experiments were limited to a binary prediction task (predicting whether a turn was a correction or a non-correction) and only considered features readily available to our dialogue system. In this section we present additional experiments removing some of these restrictions, with the goal of further investigating some of the descriptive findings discussed in Section 3.

Recall from Section 3.2 that there were some differences in the prosodic features of corrections produced by native versus non-native speakers when such features were normalized by the first turn in the dialogue. We thus investigated whether adding a native speaker feature (currently manually labeled) to the prosodic feature set Norm1 (DIA = all) would improve prediction accuracy. Although the error was reduced from 24.32% to 22.68%, this difference was not statistically significant. Furthermore, when we added the native speaker feature to both the best-performing ruleset in Table 7 (Raw+ASR+SYS+POS, DIA = PreTurn) and the best-performing prosodic feature set (Raw, DIA = all), the error rates actually increased; again, however, the differences were not statistically significant.

Also, in Sections 3.1 and 3.4, we identified differences between different types of corrections, which suggests that our features might be more effectively used to predict each correction type differently. In other words, what would happen if instead of predicting whether a turn was a correction (T) or not (F) (the binary classification task investigated above), we predicted whether a turn was ADD, ADD/OMIT, OMIT, PAR, REP, or F (i.e., not a correction). Because, as Table 2 shows, we only have limited amounts of data for

if (dur $\geq$ 1.322) $\wedge$ (pmndur $\geq$ 2.10576) **then** T (290/91)
if (pmndur $\geq$ 1.121814) $\wedge$ (dur $\geq$ 1.21814) $\wedge$ (zeros $\leq$ 0.569767) $\wedge$ (rmsmax $\geq$ 1350.81) $\wedge$ (pretempo $\leq$ 2.34637) **then** T (39/3)
if (dur $\geq$ 0.66384) $\wedge$ (pmndur $\geq$ 1.20889) $\wedge$ (tempo $\geq$ 2.90934) $\wedge$ (pmnppau $\geq$ 0.823703) **then** T (90/64)
else F (1495/256)

Figure 6
Ruleset for raw prosodic features (Raw, DIA = all).

several of our classes (e.g., only 2% of our corrections are ADD/OMIT); we performed a simpler version of this experiment, combining our three lowest frequency classes (ADD, ADD/OMIT, and PAR) into the single class MISC.

Using the best feature set from Table 7 (Raw+ASR+SYS+POS, DIA = PreTurn), Table 8 shows our results using 25-fold cross-validation. First, note that our overall estimated error is now 24.13% $\pm$ 0.89%. Although this is a huge increase compared to the 15.72% error rate of our original binary classifier, it should be noted that considering correction types separately makes our class distribution quite skewed, with the data for our three correction classes much smaller than the majority class. Nevertheless, our classifier yields a slight but significant decrease compared to the majority baseline error, and a nonsignificant decrease compared to the Prerejbool baseline error (both baselines remain the same as in Table 7). With respect to precision and recall, although the absolute numbers for corrections are much lower than in Table 7, we again see that predicting corrections yields higher precision than recall, whereas predicting non-corrections yields higher recall than precision. Finally, an examination of the learned ruleset (which contains four rules for predicting MISC, two rules for predicting OMIT, and seven rules for predicting REP) does show that features are used differently across correction types. For example, the feature **prestr** is only used to predict repetition corrections (in particular, after a turn containing *help*). Our rules also show some overlap with our earlier descriptive findings. For example, we noted that corrections of rejections were more likely to be repetitions, and find the feature **prerejbool** in two of the rules for predicting repetitions. These findings suggest that if more data were available, predicting corrections by type might prove a useful strategy.

5. Conclusions

In this article we have presented results of an analysis of corrections in the TOOT spoken dialogue corpus. We first introduced the TOOT spoken dialogue corpus and our labeling scheme to identify different types of corrections. The TOOT corpus is representative of many current research and commercial dialog systems in focusing on the travel domain. Also, since data were collected using a variety of dialog strategies with different types of initiative and confirmation, results obtained with this corpus are more likely to have general usefulness for builders of other spoken dialogue systems.

We next presented a statistical description of the corrections we labeled. In general, it appears that corrections are a serious problem for ASR, being recognized much more poorly than non-corrections but not being rejected any more frequently. Some corrections types are more difficult to handle for systems than others, with repetitions

Table 8
Predicting correction types (error $\pm$ SE = 24.13 $\pm$ 0.89)

Class	Recall	Precision	$F_\beta = 1$
FALSE	93.30	82.37	.87
REP	33.86	56.00	.41
MISC	36.17	48.36	.38
OMIT	25.47	50.13	.32

and corrections that omit information from the original turn being much better recognized than corrections that add or paraphrase such information. Confirming previous studies of repetition corrections, we found that corrections in general differ from non-corrections prosodically: They are higher in f0, softer, longer, follow longer pauses, and contain less internal silence than non-corrections. Also, corrections more distant from the error they are correcting are louder, higher in pitch, longer, slower, and follow longer pauses than closer corrections. Both findings suggest a correlation between corrections and hyperarticulation; however, most prosodic differences persist even when perceptually hyperarticulated turns are removed from the sample, and perceptual hyperarticulation is not significantly correlated with distance from original error. We hypothesize that recognizers may be more sensitive to hyperarticulatory tendencies than humans.

The second part of this article discusses results of machine-learning experiments designed to evaluate how well we can distinguish user corrections from non-corrections using features automatically available to dialogue systems. Clearly, new techniques must be developed to interpret such corrections, but such techniques can only be effective if corrections can be reliably identified as such for special handling. Using a large set of prosodic, ASR-derived, and system-specific features, both for the current turn and for contextual windows, and using summary features of the prior dialogue, we have demonstrated that it is possible to classify user corrections significantly better than either of two baseline classifiers (15.72% error vs. 25.70–28.99%). More usefully perhaps for current spoken dialogue systems, we have found that we can derive classifiers that perform equivalently well using only features currently available to most speech recognizers, such as acoustic confidence score, recognized string, grammar, and features easily derived from these data. For example, using only such features, we can classify user corrections with an estimated success rate of 16.41%. So, it does, in fact, seem quite feasible for current systems to identify user corrections using data they typically do not make use of.

Given that our findings show that corrections can be classified well using quite distinct feature sets, a possible future line of research would be to try classification combination schemes. For instance, one could envision a form of metalearning or boosting that combines classifications using different feature sets (e.g., ASR vs. prosodic vs. context), or that combines the output of different learning algorithms (e.g., Ripper combined with memory-based learning; see, e.g., Lendvai 2004). Kirchhoff (2001) presents some results of classifier combination schemes, showing some improvements in detection of corrections when using cascading, but especially when using boosting.

The next steps, developing techniques to interpret these turns more accurately and to use correction prediction to drive modifications in dialogue strategy, are both subjects of our future research. Also, whereas our analyses so far have given us overall information about the relative contribution of various feature sets for the automatic classification of corrections, one interesting problem for the future is to get more specific information about the cues that characterize corrections, especially for the development of on-line error-correction detection. In this respect, an interesting observation has been made by Kirchhoff (2001), who reports that correction classification using only features of the first half of a turn performs equally well as a classification using features of the turn as a whole; this could be explained by the fact that speakers tend to put characteristic cue phrases, such as "no" or "help," in the beginning of a turn. Additional research is needed to find strategies that use the detection of corrections to look back in the dialogue history to identify the utterance being corrected or even the actual problematic words in these turns. Finally, it would be worthwhile to investigate speaker-specific

correction strategies in more detail, the possible effect on such strategies of the user's experience with a system, and his or her linguistic background.

Acknowledgments

Marc Swerts is also affiliated with the University of Antwerp. His research is sponsored by the Netherlands Organisation for Scientific Research (NWO). This work was performed when the authors were at AT&T Labs—Research.

References

Ammicht, E., A. Potamianos, and E. Fosler-Lussier. 2001. Ambiguity representation and resolution in spoken dialogue systems. In *Proceedings of EUROSPEECH-01*, pages 2217–2220, Aalborg.

Andorno, M., P. Laface, and R. Gemello. 2002. Experiments in confidence scoring for word and sentence verification. In *Proceedings of International Conference on Spoken Language Processing-02*, pages 1377–1381, Denver.

Barkhuysen, P., E. Krahmer, and M. Swerts. 2005. Problem detection in human–machine interactions based on facial expressions of users. *Speech Communication*, 45:343–359.

Batliner, A., K. Fischer, R. Huber, J. Spilker, and E. Nöth. 2003. How to find trouble in communication. *Speech Communication*, 40:117–143.

Bell, L. and J. Gustafson. 1999. Repetition and its phonetic realizations: Investigating a Swedish database of spontaneous computer-directed speech. In *Proceedings of International Congress of Phonetic Sciences-99*, pages 1221–1224, San Francisco.

Bouwman, A. G., J. Sturm, and L. Boves. 1999. Incorporating confidence measures in the Dutch train timetable information system developed in the ARISE project. In *Proceedings International Conference on Acoustics, Speech and Signal Processing*, volume 1, pages 493–496, Phoenix.

Bulyko, I., K. Kirchhoff, M. Ostendorf, and J. Goldberg. 2005. Error-correction detection and response generation in a spoken dialogue system. *Speech Communication*, 45:271–288.

Cohen, P. 1995. *Empirical Methods for Artificial Intelligence*. MIT Press, Boston.

Cohen, W. 1996. Learning trees and rules with set-valued features. In *14th Conference of the American Association of Artificial Intelligence, AAAI*, pages 709–716, Portland.

Falavigna, D., R. Gretter, and G. Riccardi. 2002. Acoustic and word lattice based algorithms for confidence scores. In *Proceedings of International Conference on Spoken Language Processing-02*, pages 1621–1624, Denver.

Guillevic, D., S. Gandrabur, and Y. Normandin. 2002. Robust semantic confidence scoring. In *Proceedings of International Conference on Spoken Language Processing-02*, pages 853–856, Denver.

Hirschberg, J., D. Litman, and M. Swerts. 1999. Prosodic cues to recognition errors. In *Proceedings of the Automatic Speech Recognition and Understanding Workshop (ASRU'99)*, pages 349–352, Keystone.

Hirschberg, J., D. Litman, and M. Swerts. 2004. Prosodic and other cues to speech recognition failures. *Speech Communication*, 43:155–175.

Kamm, C., S. Narayanan, D. Dutton, and R. Ritenour. 1997. Evaluating spoken dialog systems for telecommunication services. In *Proceedings of EUROSPEECH-97*, pages 2203–2206, Rhodes.

Karsenty, L. and V. Botherel. 2005. Transparency strategies to help users handle system errors. *Speech Communication*, 45:305–324.

Kirchhoff, Katrin. 2001. A comparison of classification techniques for the automatic detection of error corrections in human–computer dialogues. In *Proceedings of the NAACL Workshop on Adaptation in Dialogue Systems*, pages 33–40, Pittsburgh, PA.

Krahmer, E., M. Swerts, M. Theune, and M. Weegels. 2001. Error detection in spoken human–machine interaction. *International Journal of Speech Technology*, 4(1):19–30.

Levow, Gina-Anne. 1998. Characterizing and recognizing spoken corrections in human–computer dialogue. In *Proceedings of the 36th Annual Meeting of the Association of Computational Linguistics, COLING/ACL 98*, pages 736–742, Montreal.

Lendvai, Piroska. 2004. Extracting information from spoken user input. A machine-learning approach. Unpublished Ph.D. dissertation, Tilburg University.

Litman, D. and S. Pan. 1999. Empirically evaluating an adaptable spoken

dialogue system. In *Proceedings of the 7th International Conference on User Modeling (UM)*, pages 55–64, Banff.
Litman, D., M. Walker, and M. Kearns. 1999. Automatic detection of poor speech recognition at the dialogue level. In *Proceedings of the 37th Annual Meeting of the Association of Computational Linguistics, ACL99*, pages 309–316, College Park.
McTear, M., I. A. O'Neill, P. Hanna, and X. Liu. 2005. Handling errors and determining confirmation strategies—an object-based approach. *Speech Communication*, 45:249–269.
Moreno, P. J., B. Logan, and B. Raj. 2001. A boosting approach for confidence scoring. In *Proceedings of EUROSPEECH-01*, pages 2109–2112, Aalborg.
Oviatt, S. L., G. Levow, M. MacEarchern, and K. Kuhn. 1996. Modeling hyperarticulate speech during human-computer error resolution. In *Proceedings of International Conference on Spoken Language Processing-96*, pages 801–804, Philadelphia.
Prodanov, P. and A. Drygajlo. 2005. Bayesian networks based multimodality fusion for error handling in human–robot dialogues under noisy conditions. *Speech Communication*, 45:231–248.
Sharp, R. D., E. Bocchieri, C. Castillo, S. Parthasarathy, C. Rath, M. Riley, and J. Rowland. 1997. The Watson speech recognition engine. In *Proceedings International Conference on Acoustics, Speech and Signal Processing97*, pages 4065–4068, Munich.
Shimojima, A., Y. Katagiri, H. Koiso, and M. Swerts. 2001. An experimental study on the informational and grounding functions of prosodic features of Japanese echoic responses. *Speech Communication*, 43:155–175.
Skantze, G. 2005. Exploring human error recovery strategies: Implications for spoken dialogue systems. *Speech Communication*, 45:325–341.
Soltau, Hagen and Alex Waibel. 1998. On the influence of hyperarticulated speech on recognition performance. In *Proceedings of International Conference on Spoken Language Processing-98*, pages 225–228, Sydney.
Soltau, Hagen and Alex Waibel. 2000. Specialized acoustic models for hyperarticulated speech. In *Proceedings of International Conference on Acoustics, Speech and Signal Processing 2000*, pages 1779–1782, Istanbul.
Soltau, H., H. Metze, and A. Waibel. 2002. Compensating for hyperarticulation by modeling articulatory properties. In *Proceedings of International Conference on Spoken Language Processing-02*, pages 83–86, Denver.
Sturm, J. and L. Boves. 2005. Effective error recovery strategies for multimodal form-filling applications. *Speech Communication*, 45:289–303.
Swerts, M., D. Litman, and J. Hirschberg. 2000. Corrections in spoken dialogue systems. In *Proceedings of International Conference on Spoken Language Processing-00*, pages 615–618, Beijing.
Talkin, D. 1995. A Robust algorithm for pitch tracking (RAPT). In W. B. Klein and K. K. Paliwal, editors, *Speech Coding and Synthesis*. Elsevier Science, Athens, pages 495–518.
Torres, F., L. Hurtado, F. García, E. Sanchis, and E. Segarra. 2005. Error handling in a stochastic dialog system through confidence measures. *Speech Communication*, 45:211–229.
Wade, E., E. E. Shriberg, and P. J. Price. 1992. User behaviors affecting speech recognition. In *Proceedings of International Conference on Spoken Language Processing-92*, volume 2, pages 995–998, Banff.
Walker, M., I. Langkilde, J. Wright, A. Gorin, and D. Litman. 2000. Learning to predict problematic situations in a spoken dialogue system: Experiments with How may I help you? In *Proceedings of NAACL-00*, pages 210–217, Seattle.
Wang, H.-M. and Y.-C. Lin. 2002. Error-tolerant spoken language understanding with confidence measuring. In *Proceedings of International Conference on Spoken Language Processing-02*, pages 1625–1628, Denver.
Zhang, R. and A. Rudnicky. 2001. Word level confidence annotation using combinations of features. In *Proceedings of EUROSPEECH-01*, pages 2105–2108, Aalborg.