Identifying User Corrections Automatically in Spoken Dialogue Systems

Julia Hirschberg[1] and **Diane Litman**[1] and **Marc Swerts**[2]
[1]AT&T Labs–Research, Florham Park, NJ, USA
[2]IPO, Eindhoven, The Netherlands, and CNTS, Antwerp, Belgium
{julia/diane}@research.att.com,m.g.j.swerts@tue.nl

Abstract

We present results of machine learning experiments designed to identify user corrections of speech recognition errors in a corpus collected from a train information spoken dialogue system. 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

Users' evaluations of spoken dialogue systems largely depend on the number of errors the system makes (Walker et al., 2000a) and how easy it is for the user and system to correct them. Poor automatic speech recognition (ASR) accuracy is the primary source of error in most systems, compounded by user behavior when confronted by such error. Studies have shown that speakers tend to switch to a prosodically 'marked' speaking style — **hyperarticulated speech**, e.g. *I said BAL-TI-MORE, not Boston* — after ASR errors (Wade et al., 1992; Oviatt et al., 1996; Levow, 1998; Bell and Gustafson, 1999). While possibly effective in human-human communicative settings, such correction behavior appears to lead to further errors in human-machine interactions (Levow, 1998; Soltau and Waibel, 2000), perhaps because it differs from the speech most recognizers are trained on. In addition, system responses indicating incorrect beliefs, such as an implicit verification containing mistaken information (e.g. *Where do you want to go from Boston* when the user has said she wants to depart from Baltimore), cause considerable difficulties for users faced with the need to correct a misconception and also answer a new question (Krahmer et al., 1999).

To date, attempts to improve system performance have largely focussed on improving ASR accuracy, or simplifying the task, either by further constraining the domain and functionality of the system or further restricting the vocabulary the system must recognize. 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.

Ideally, such repair strategies, 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. In previous research (Litman et al., 2000; Hirschberg et al., 2000), 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). 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.

Now we have turned to the question of how users *correct* such errors. In a descriptive analysis of user corrections in the TOOT corpus (Swerts et al., 2000), we found that corrections differ significantly from non-corrections prosodically, being higher in pitch, louder, longer, with longer pauses preceding them and less internal silence. They are also misrecognized more frequently than non-corrections — though 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 versus paraphras-

ing misrecognized information).

These findings suggest 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 which 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 the current study, 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. Our domain is the TOOT spoken dialogue corpus, which we describe in Section 2. In Section 3, we describe the features we use for our machine learning experiments. Section 4 presents the results of those experiments. In Section 5 we summarize our conclusions and describe future research directions.

2 The TOOT Corpus

Our corpus consists of 152 dialogues between human subjects and TOOT, a spoken dialogue system that allowed users access to train information from the web via telephone. The TOOT corpus was collected in a laboratory setting, to study variations in dialogue strategy and in user-adapted interaction (Litman and Pan, 1999). TOOT was implemented using an interactive voice response platform developed at AT&T, combining ASR and text-to-speech with a phone interface (Kamm et al., 1997). The system's speech recognizer was a speaker-independent hidden Markov model system, with context-dependent phone models for telephone speech and constrained grammars defining the vocabulary allowed at any dialogue state. The platform supported barge-in. This system rejected a turn whenever its acoustic confidence score fell below thresholds predefined for each dialogue state.

Subjects were 39 student interns; 20 native speakers and 19 non-native, 16 female and 23 male. They were asked to perform four tasks using 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 versus non-adaptive). Dialogues were recorded and system and user behavior logged automatically.

We examined 2328 user turns from 152 dialogues, with a mean number of turns per dialogue of 15.3. The turns were transcribed and automatically compared to the recognized string to produce a *word error rate* (WER) for each turn; the mean WER over all turns was 40%. Actual words per turn averaged 3.6 and recognized words per turn, 2.4. The Concept Accuracy (CA) for 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. CA errors and explicit system rejections were the only errors users could identify as such, and so were the only ones corrected.

In addition, two authors independently labeled each turn as to whether or not it constituted a correction of a prior system failure. They also identified the turn being corrected, and the type of each correction: REP (repetition, including differences in pronunciation or fluency), PAR (paraphrase); ADD (task-relevant content added); OMIT (content omitted); and ADD/OMIT (content both added and omitted). Labeler disagreement was subsequently resolved by consensus. 675 of the 2328 user turns in the corpus (29%) were judged to be corrections. A sample interaction between TOOT and a user, containing the different correction types and their labels is given in Figure 1.

3 Characterizing Corrections

In this section we describe the features used in the machine learning experiments described in Section 4 and the motivation behind their selection. The entire feature set is presented in Figure 2, and includes only features that could be automatically available to a dialogue system.

3.1 Prosodic Features

A previous study (Swerts et al., 2000) 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

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

Table 1: Means and Standard Deviations for Prosodic Features over All Turns

	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:	8:00 a.m.	780	NA	NA
S:	*Do you want to leave around 10:00 p.m.*			
U:	8:00	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:	8:00 a.m.	783	781	ADD

Figure 1: Dialogue Fragment with Correction Labels

(**zeros**).[1] We examined both raw and normalized versions of these features, with normalization either by first turn (**norm1**) or by prior turn (**norm2**) in task. F0 and rms values were calculated from the output of Entropic Research Laboratory's pitch tracker, *get_f0*. Speaking rate (**tempo**) was approximated from syllables per second in the recognized string, a method we previously found to accurately approximate actual rate. Turn-internal silence (**zeros**) was defined as the percentage of zero frames in the turn, i.e., roughly the percentage of time the speaker was silent. Table 1 shows the overall means and standard deviations for these features over the corpus.

[1] While most features were automatically computed, **dur** and **ppau** were hand segmented, as the turn-level files created by TOOT were not available; these would have been available from the turn-level speech, but perhaps not with the same accuracy.

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., priorynstrnum/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 2: Feature Set for Predicting Corrections

3.2 ASR Features

Since corrections in our corpus were misrecognized more frequently than non-corrections (Swerts et al., 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 recognizer's best hypothesis (**str**), and the turn-level acoustic confidence score it produced (**conf**).[2] 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

[2] Confidence scores ranged from -0.087662 to -9.884418.

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**).

3.3 System Experimental Features

In (Swerts et al., 2000) we found 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**).

3.4 Dialogue Position and History Features

(Swerts et al., 2000) 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 et al., 1999; Walker et al., 2000b) 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.[3] 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 Predicting Corrections

In this section we investigate whether the features described in Section 3 (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 2 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 in Section 2. 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 2 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 2328 turns in our corpus. The 'Features' column identifies the set of features (as defined in Figure 2) used to learn the model. The second column, 'DIA', indicates which type of dialogue history features (PreTurn, PrepreTurn, Prior, 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.[4] For comparison purposes,

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

[4] *Recall* is the percentage of actual members of a class that are identified, while *precision* is the percentage of predicted class members that are in fact members. The definition of F_β is $\frac{(\beta^2+1)PrecisionRecall}{\beta^2 Precicison+Recall}$; $\beta = 1$ equally weights precision and recall. These values are computed using our own cross-

			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								

Table 2: Estimated Error, Recall, Precision, and $F_\beta = 1$ for Predicting Corrections

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 — 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 2 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 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.[5] In addition, the recall, precision and $F_\beta = 1$ values in Table 2 show that corrections are generally predicted with better precision than recall while 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 2 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

validation program, while error is computed using RIPPER's cross-validation option.

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

corrections successfully, using various combinations of feature types. This finding is an important one, since it suggests that systems which 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 2 shows that use of only non-prosodic features (ASR+SYS+POS, DIA = all, Error = 17.01%) 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. While 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 dialogue 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% versus 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 2 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 which are easily made available to current spoken dialogue systems.

The classification model learned from the best performing feature set in Table 2 is shown in Figure 3. 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 $\leq$ -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

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

Figure 3: Best Performing Ruleset (Raw+ASR+SYS+POS, DIA=PreTurn)

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, while **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 4). For example, the fourth rule predicts that the current turn

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

Figure 4: Ruleset for Non-Prosodic Features (ASR+SYS+POS, DIA=all)

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 3), 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**, **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 2) is shown in Figure 5. This ruleset is notably terser than those shown in Figures 3 and 4 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. Like

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 5: Ruleset for Raw Prosodic Features (Raw, DIA=all)

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.

5 Discussion

In this paper we have presented results of machine learning experiments designed to distinguish user corrections from non-corrections in the TOOT spoken dialogue corpus. Previous studies have shown that user corrections represent a serious problem for recognition in spoken dialogue system. They are recognized much more poorly than non-corrections but are *not* recognized by the system as likely to have been misrecognized, in corresponding proportion. 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 versus 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 this 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 currently do not make use of. 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.

Acknowledgments

Thanks to Walter Daelemans for his cross-validation software.

References

L. Bell and J. Gustafson. 1999. Repetition and its phonetic realizations: Investigating a Swedish database of spontaneous computer-directed speech. In *Proc. ICPhS-99*, San Francisco.

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

W. Cohen. 1996. Learning trees and rules with set-valued features. In *AAAI–96*.

J. B. Hirschberg, D. J. Litman, and M. Swerts. 2000. Generalizing prosodic prediction of speech recognition errors. In *Proc. ICSLP–00*, Beijing.

C. Kamm, S. Narayanan, D. Dutton, and R. Ritenour. 1997. Evaluating spoken dialog systems for telecommunication services. In *Proc. EUROSPEECH-97*, Rhodes.

E. Krahmer, M. Swerts, M. Theune, and M. Weegels. 1999. Error spotting in human-machine interactions. In *Proc. EUROSPEECH-99*.

G. Levow. 1998. Characterizing and recognizing spoken corrections in human-computer dialogue. In *Proc. COLING/ACL–98*.

D. J. Litman and S. Pan. 1999. Empirically evaluating an adaptable spoken dialogue system. In *Proc. 7th Int'l Conference on User Modeling*.

D. J. Litman, M. A. Walker, and Michael J. Kearns. 1999. Automatic detection of poor speech recognition at the dialogue level. In *Proc. ACL–99*.

D. J. Litman, J. B. Hirschberg, and M. Swerts. 2000. Predicting automatic speech recognition performance using prosodic cues. In *Proc. NAACL–00*, Seattle.

S. L. Oviatt, G. Levow, M. MacEarchern, and K. Kuhn. 1996. Modeling hyperarticulate speech during human-computer error resolution. In *Proc. ICSLP-96*, Philadelphia.

H. Soltau and A. Waibel. 2000. Specialized acoustic models for hyperarticulated speech. In *Proc. ICASSP–00*, Istanbul.

M. Swerts, D. Litman, and J. Hirschberg. 2000. Corrections in spoken dialogue systems. In *Proc. ICSLP–00*, Beijing.

E. Wade, E. E. Shriberg, and P. J. Price. 1992. User behaviors affecting speech recognition. In *Proc. ICSLP–92*, Banff.

M. Walker, C. Kamm, and D. Litman. 2000a. Towards developing general models of usability with PARADISE. *Natural Language Engineering*, 6, October.

M. A. Walker, I. Langkilde, J. Wright, A. Gorin, and D. Litman. 2000b. Learning to predict problematic situations in a spoken dialogue system: Experiments with how may i help you? In *Proc. NAACL–00*, Seattle.