Predicting User Reactions to System Error

Diane Litman and Julia Hirschberg
AT&T Labs–Research
Florham Park, NJ, 07932 USA
{diane/julia}@research.att.com

Marc Swerts
IPO, Eindhoven, The Netherlands,
and CNTS, Antwerp, Belgium
m.g.j.swerts@tue.nl

Abstract

This paper focuses on the analysis and prediction of so-called *aware sites*, defined as turns where a user of a spoken dialogue system first becomes aware that the system has made a speech recognition error. We describe statistical comparisons of features of these aware sites in a train timetable spoken dialogue corpus, which reveal significant prosodic differences between such turns, compared with turns that 'correct' speech recognition errors as well as with 'normal' turns that are neither aware sites nor corrections. We then present machine learning results in which we show how prosodic features in combination with other automatically available features can predict whether or not a user turn was a normal turn, a correction, and/or an aware site.

1 Introduction

This paper describes new results in our continuing investigation of prosodic information as a potential resource for error recovery in interactions between a user and a spoken dialogue system. In human-human interaction, dialogue partners apply sophisticated strategies to detect and correct communication failures so that errors of recognition and understanding rarely lead to a complete breakdown of the interaction (Clark and Wilkes-Gibbs, 1986). In particular, various studies have shown that prosody is an important cue in avoiding such breakdown, e.g. (Shimojima et al., 1999). Human-machine interactions between a user and a spoken dialogue system (SDS) exhibit more frequent communication breakdowns, due mainly to errors in the Automatic Speech Recognition (ASR) component of these systems. In such interactions, however, there is also evidence showing prosodic information may be used as a resource for error recovery. In previous work, we identified new procedures to *detect* recognition errors. In particular, we found that prosodic features, in combination with other information already available to the recognizer, can distinguish user turns that are *misrecognized* by the system far better than traditional methods used in ASR rejection (Litman et al., 2000; Hirschberg et al., 2000). We also found that user *corrections* of system misrecognitions exhibit certain typical prosodic features, which can be used to identify such turns (Swerts et al., 2000; Hirschberg et al., 2001). These findings are consistent with previous research showing that corrections tend to be *hyperarticulated* — higher, louder, longer …than other turns (Wade et al., 1992; Oviatt et al., 1996; Levow, 1998; Bell and Gustafson, 1999).

In the current study, we focus on another turn category that is potentially useful in error handling. In particular, we examine what we term *aware sites* — turns where a user, while interacting with a machine, first becomes aware that the system has misrecognized a previous user turn. Note that such aware sites may or may not also be corrections (another type of post-misrecognition turn), since a user may not immediately provide correcting information. We will refer to turns that are both aware sites and corrections as *corr-awares*, to turns that are only corrections as *corrs*, to turns that are only aware sites as *awares*, and to turns that are neither aware sites nor corrections as *norm*.

We believe that it would be useful for the dialogue manager in an SDS to be able to detect aware sites for several reasons. First, if aware sites are detectable, they can function as backward-looking error-signaling devices, making it clear to the system that something has gone wrong in the preceding context, so that, for example, the system can reprompt for information. In this way, they are similar to what others have termed 'go-back' signals (Krahmer et al., 1999). Second, aware sites can be used as forward-looking signals, indicating upcoming corrections or more drastic changes in user behavior, such as complete restarts of the task. Given that, in current systems, both corrections and restarts often lead to recognition error (Swerts et al., 2000), aware sites may be useful in preparing systems to deal with such problems.

In this paper, we investigate whether aware sites share acoustic properties that set them apart from normal turns, from corrections, and from turns which are both aware sites and corrections. We also want to test whether these different turn categories can be distinguished automatically, via their prosodic features or from other features known to or automatically detectible by a spoken dialogue system. Our domain is the TOOT spoken dialogue corpus, which we describe in Section 2. In Section 3, we present some descriptive findings on different turn categories in TOOT. Section 4 presents results of our machine learning experiments on distinguishing the different turn classes. In Section 5 we summarize our conclusions.

2 Data

The TOOT corpus was collected using an experimental SDS developed for the purpose of comparing differences in dialogue strategy. It provides access to train information over the phone and is implemented using an internal platform combining ASR, text-to-speech, a phone interface, and modules for specifying a finite-state dialogue manager, and application functions. Subjects performed four tasks with versions of TOOT, which varied confirmation type and locus of initiative (system initiative with explicit system confirmation, user initiative with no system confirmation until the end of the task, mixed initiative with implicit system confirmation), as well as whether the user could change versions at will using voice commands. Subjects were 39 students, 20 native speakers of standard American English and 19 non-native speakers; 16 subjects were female and 23 male. The exchanges were recorded and the system and user behavior logged automatically. Dialogues were manually transcribed and user turns automatically compared to the corresponding ASR (one-best) recognized string to produce a *word accuracy* score (WA) for each turn. Each turn's *concept accuracy* (CA) was labeled by the experimenters from the dialogue recordings and the system log; if the recognizer correctly captured all the task-related information given in the user's original input (e.g. date, time, departure or arrival cities), the turn was given a CA score of 1, indicating a *semantically correct recognition.* Otherwise, the CA score reflected the percentage of correctly recognized task concepts in the turn. For the study described below, we examined 2328 user turns from 152 dialogues generated during these experiments. 194 of the 2320 turns were rejected by the system.

To identify the different turn categories in the corpus, two authors independently labeled each turn as to whether or not it constituted a correction of a prior system failure (a CA error or a rejection) and what turn was being corrected, and whether or not it represented an aware site for a prior failure, and, if so, the turn which the system had failed on. Labeler disagreement was subsequently resolved by consensus. The fragment in Figure 1, produced with a version of TOOT in which the user has the initiative with no confirmation until the end of the task, illustrates these labels. This example illustrates cases of *corraware*, in which both the user's awareness and correction of a misrecognition occur in the same turn (e.g. turns 1159 and 1160, after system prompts for information already given in turn 1158). It also illustrates cases in which aware sites and corrections occur in different turns. For example, after the immediate explicit system confirmation of turn 1162, the user first becomes aware of the system errors (turn 1163), then separately corrects them (turn 1164); turn 1163 is thus an *aware* turn and turn 1164 a *corr*. When no immediate confirmation of an utterance occurs (as with turn 1158), it may take several turns before the user becomes aware of any

Turn		Turn ID	Aware of	Corr of	Type
S:	*How may I help you?*				
U:	Get me the train to New York City from Baltimore on Sunday at eight thirty pm	1158	NA	NA	NORM
S:	*Which city do you want to go to?*				
U:	New York City	1159	1158	1158	CORRAWARE
S:	*Which city do you want to leave from?*				
U:	Baltimore	1160	1158	1158	CORRAWARE
S:	*Do you want me to find the trains from Baltimore to New York City today at anytime now?*				
U:	No	1161	1158	NA	AWARE
S:	*How may I help you?*				
U:	Get me the train to New York City from Baltimore on Sunday at eight thirty pm	1162	NA	1158	CORR
S:	*Do you want me to find the trains from Baltimore to New York City today at anytime now?*				
U:	No	1163	1162	NA	AWARE
S:	*How may I help you?*				
U:	Get me the train on Sunday at eight thirty pm	1164	NA	1162	CORR

Figure 1: Dialogue Fragment with Aware and Correction Labels.

misrecognition errors. For example, it is not until turn 1161 that the user first becomes aware of the error in date and time from 1158; the user then corrects the error in 1162. So, 1161 is classified as an aware and 1162 as a corr. Note that corr turns represent 13% of the turns in our corpus, awares represent 14%, corrawares account for 16%, and norm turns represent 57% of the turns in the corpus.

3 Descriptive Analysis and Results

We examined prosodic features for each user turn which had previously been shown to be useful for predicting misrecognized turns and corrections:[1] maximum and mean fundamental frequency values (F0 Max, F0 Mean), maximum and mean energy values (RMS Max, RMS Mean), total duration (Dur), length of pause preceding the turn (Ppau), speaking rate (Tempo) and amount of silence within the turn (%Sil). 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*, with no post-correction. Timing variation was represented by four features. Duration within and length of pause between turns was computed from the temporal labels associated with each turn's beginning and end. Speaking rate was approximated in terms of syllables in the recognized string per second, while %Sil was defined as the percentage of zero frames in the turn, i.e., roughly the percentage of time within the turn that the speaker was silent.

To see whether the different turn categories were prosodically distinct from one another, we applied the following procedure. We first calculated mean values for each prosodic feature for each of the four turn categories produced by each individual speaker. So, for speaker A, we divided all turns produced into four classes. For each class, we then calculated mean F0 Max, mean F0 Mean, and so on. After this step had been repeated for each speaker and for each feature, we then created four vectors of speaker means for each individual prosodic feature. Then, for each prosodic feature, we ran a one-factor within subjects anova on the means to learn whether there was an overall effect of turn category.

Table 1 shows that, overall, the turn categories do indeed differ significantly with respect to different prosodic features; there is a significant, overall effect of category on F0 Max, RMS Max, RMS Mean, Duration, Tempo and %Sil. To identify which pairs of turns were significantly different where there was an overall significant effect, we performed posthoc paired t-tests using the Bonferroni method to adjust the p-level to 0.008 (on the basis of the number of possible pairs that

[1] While the features were automatically computed, beginnings and endings were hand segmented from recordings of the entire dialogue, as the turn-level speech files used as input in the original recognition process created by TOOT were unavailable.

	Turn categories				
Feature	Normal	Correction	Aware	Corraware	F-stat
***F0 Max (Hz)	220.05	263.40	216.87	229.00	$F_{(3,96)}$=10.477
F0 Mean (Hz)	161.78	173.43	162.61	158.24	$F_{(3,96)}$=1.575
***RMS Max (dB)	1484.14	1833.62	1538.91	1925.38	$F_{(3,96)}$=7.548
*RMS Mean (dB)	372.47	379.65	425.96	464.16	$F_{(3,96)}$=3.190
***Dur (sec)	1.43	4.39	1.12	2.33	$F_{(3,99)}$=34.418
Ppau (sec)	0.60	0.93	0.87	0.80	$F_{(3,99)}$=1.325
**Tempo (syls/sec)	2.59	2.38	2.16	2.43	$F_{(3,99)}$=4.206
*%Sil (sec)	0.46	0.41	0.44	0.42	$F_{(3,96)}$=3.182
Significance level: *(p<.05), **(p<.01), ***(p<.001)					

Table 1: Mean Values of Prosodic Features for Turn Categories.

	Prosodic features							
Classes	F0 max	F0 mean	RMS max	RMS mean	Dur	Ppau	Tempo	%Sil
norm/corr	–		–		–			+
norm/aware					+			
norm/corraware			–	–				
aware/corr	–		–		–		–	
aware/corraware	–		–		–			
corraware/corr	–				–			

Table 2: Pairwise Comparisons of Different Turn Categories by Prosodic Feature.

can be drawn from an array of 4 means). Results are summarized in Table 2, where ' + ' or ' – ' indicates that the feature value of the first category is either significantly higher or lower than the second. Note that, for each of the pairs, there is at least one prosodic feature that distinguishes the categories significantly, though it is clear that some pairs, like aware vs. corr and norm vs. corr appear to have more distinguishing features than others, like norm vs. aware. It is also interesting to see that the three types of post-error turns are indeed prosodically different: awares are less prominent in terms of F0 and RMS maximum than corrawares, which, in turn, are less prominent than corrections, for example. In fact, awares, except for duration, are prosodically similar to normal turns.

4 Predictive Results

We next wanted to determine whether the prosodic features described above could, alone or in combination with other automatically available features, be used to predict our turn categories automatically. This section describes experiments using the machine learning program RIPPER (Cohen, 1996) to automatically induce prediction models from our data. Like many learning programs, RIPPER takes as input the classes to be learned, a set of feature names and possible values, and training data specifying the class and feature values for each training example. RIPPER outputs a classification model for predicting the class of future examples, expressed as an ordered set of if-then rules. The main advantages of RIPPER for our experiments are that RIPPER supports "set-valued" features (which allows us to represent the speech recognizer's best hypothesis as a set of words), and that rule output is an intuitive way to gain insight into our data.

In the current experiments, we used 10-fold cross-validation to estimate the accuracy of the rulesets learned. Our predicted classes correspond to the turn categories described in Section 2 and variations described below. We represent each user turn using the feature set shown in Figure 2, which we previously found useful for predicting corrections (Hirschberg et al., 2001). A subset of the features includes the automatically computable raw prosodic features shown in Table 1 (Raw), and normalized versions of these features, where normalization was done by first turn (Norm1) or by previous turn (Norm2) in a dialogue. The set labeled 'ASR' contains standard input and output of the speech recognition process, which grammar was used for the dialogue state the system believed the user to be in (gram),

Raw: f0 max, f0 mean, rms max, rms mean, dur, ppau, tempo, %sil;

Norm1: f0 max1, f0 mean1, rms max1, rms mean1, dur1, ppau1, tempo1, %sil1;

Norm2: f0 max2, f0 mean2, rms max2, rms mean2, dur2, ppau2, tempo2, %sil2;

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

System Experimental: inittype, conftype, adapt, realstrat;

Dialogue Position: diadist;

PreTurn: features for preceding turn (e.g., pref0max);

PrepreTurn: features for 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 feature, the mean of the feature's value over all prior turns (e.g., pmnf0max);

Figure 2: Feature Set.

the system's best hypothesis for the user input (str), and the acoustic confidence score produced by the recognizer for the turn (conf). As subcases of the str feature, we also included 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 were often used to signal problems in our system. We also derived features to approximate the length of the user turn in words (wordsstr) and in syllables (syls) from the str features. And we added a boolean feature identifying whether or not the turn had been rejected by the system (rejbool). Next, we include a set of features representing the system's dialogue strategy when each turn was produced. These include the system's current initiative and confirmation strategies (inittype, conftype), whether users could adapt the system's dialogue strategies (adapt), and the combined initiative/confirmation strategy in effect at the time of the turn (realstrat). Finally, given that our previous studies showed that preceding dialogue context may affect correction behavior (Swerts et al., 2000; Hirschberg et al., 2001), we included a feature (diadist) reflecting the distance of the current turn from the beginning of the dialogue, and a set of features summarizing aspects of the prior dialogue: for the latter features, we included both the number of times prior turns exhibited certain characteristics (e.g. priorcancnum) and the percentage of the prior dialogue containing one of these features (e.g. priorcancpct). We also examined means for all raw and normalized prosodic features and some word-based features over the entire dialogue preceding the turn to be predicted (pmn__). Finally, we examined more local contexts, including all features of the preceding turn (pre__) and for the turn preceding that (ppre__).

We provided all of the above features to the learning algorithm first to predict the four-way classification of turns into normal, aware, corr and corraware. A baseline for this classification (always predicting norm, the majority class) has a success rate of 57%. Compared to this, our features improve classification accuracy to 74.23% (+/– 0.96%). Figure 3 presents the rules learned for this classification. Of the features that appear in the ruleset, about half are features of current turn and half features of the prior context. Only once does a system feature appear, suggesting that the rules generalize beyond the experimental conditions of the data collection. Of the features specific to the current turn, prosodic features dominate, and, overall, timing features (dur and tempo especially) appear most frequently in the rules. About half of the contextual features are prosodic ones and half are ASR features, with ASR confidence score appearing to be most useful. ASR features of the current turn which appear most often are string-based features and the grammar state the system used for recognizing the turn. There appear to be no differences in which type of features are chosen to predict the different classes.

If we express the prediction results in terms of precision and recall, we see how our classification accuracy varies for the different turn categories (Table 3). From Table 3, we see that the majority class (normal) is most accurately classified. Predictions for the other three categories, which occur about equally often in our corpus, vary considerably, with modest results for corr and corraware, and rather poor results for aware. Table 4 shows a confusion matrix for the four classes, produced by

if (gram=universal) ∧ (dur2 ≥ 7.31) **then** *CORR*
if (dur2 ≥ 2.19) ∧ (priornofeatpct ≥ 0.09) ∧ (tempo ≥ 1.50) ∧ (pmntempo ≤ 2.39) **then** *CORR*
if (dur2 ≥ 1.53) ∧ (pmnwordsstr ≥ 2.06) ∧ (tempo1 ≥ 1.07) ∧ (predur ≥ 0.80) ∧ (prenofeat=F) ∧ (presyls ≤ 4) **then** *CORR*
if (predur1 ≤ 0.26) ∧ (dur ≥ 0.79) ∧ (rmsmean2 ≥ 1.51) ∧ (f0mean ≤ 173.49) **then** *CORR*
if (dur2 ≥ 1.41) ∧ (prenofeat=T) ∧ (str contains word 'eight') **then** *CORR*
if (predur1 ≤ 0.18) ∧ (dur2 ≥ 4.21) ∧ (dur1 ≤ 0.50) ∧ (f0mean ≤ 276.43) **then** *CORR*
if (predur1 ≤ 0.19) ∧ (ppregram=cityname) ∧ (rmsmax1 ≥ 1.10) ∧ (pmntempo2 ≤ 1.64) **then** *CORR*
if (realstrat=SystemImplicit) ∧ (gram=cityname) ∧ (pmnf0mean1 ≤ 0.96) **then** *CORR*
if (preconf ≤ -2.66) ∧ (dur2 ≤ 0.31) ∧ (pprenofeat=T) ∧ (tempo2 ≥ 0.61) **then** *AWARE*
if (preconf ≤ -2.85) ∧ (syls ≤ 2) ∧ (predur ≥ 1.05) ∧ (pref0max > 4.82) ∧ (tempo2 ≥ 0.58) ∧ (pmn%sil ≤ 0.53) **then** *AWARE*
if (preconf ≤ -3.34) ∧ (syls ≤ 2) ∧ (ppau ≥ 0.57) ∧ (conf ≥ -3.07) ∧ (preppau ≥ 0.72) **then** *AWARE*
if (dur ≥ 0.74) ∧ (pmndur ≥ 2.57) ∧ (preconf ≤ -4.36) ∧ (f0mean2 ≥ 0.90) **then** *CORRAWARE*
if (preconf ≤ -2.80) ∧ (pretempo ≤ 2.16) ∧ (preconf ≤ -3.95) ∧ (tempo1 ≤ 4.67) **then** *CORRAWARE*
if (preconf ≤ -2.80) ∧ (dur ≥ 0.66) ∧ (rmsmean ≥ 488.56) **then** *CORRAWARE*
if (preconf ≤ -3.56) ∧ (dur2 ≥ 0.64) ∧ (prerejbool=T) **then** *CORRAWARE*
if (pretempo ≤ 0.71) ∧ (tempo ≤ 3.31) **then** *CORRAWARE*
if (preconf ≤ -3.01) ∧ (tempo2 ≥ 0.78) ∧ (pmndur ≥ 2.83) ∧ (pmnf0mean ≥ 199.84) **then** *CORRAWARE*
if (pmnconf ≤ -3.10) ∧ (prestr contains the word 'help') ∧ (pmndur2 ≤ 2.01) ∧ (ppau ≤ 0.98) **then** *CORRAWARE*
if (pmnconf ≤ -3.10) ∧ (gram=universal) ∧ (pregram=universal) ∧ (%sil ≤ 0.39) **then** *CORRAWARE*
else *NORM*

Figure 3: Rules for Predicting 4 Turn Categories.

	Precision (%)	Recall (%)
norm	80.09	89.39
corr	72.86	61.66
aware	61.01	39.79
corraware	61.76	61.72
Accuracy: 74.23% (± 0.96%); baseline: 57%		

Table 3: 4-way Classification Performance.

applying our best ruleset to the whole corpus. This

	Classified as			
	norm	corr	aware	corraware
norm	1263	14	11	38
corr	68	219	0	7
aware	149	1	130	47
corraware	53	5	8	315

Table 4: Confusion Matrix, 4-way Classification.

matrix clearly shows a tendency for the minority classes, aware, corr and corraware, to be falsely classified as normal. It also shows that aware and corraware are more often confused than the other categories.

These confusability results motivated us to collapse the aware and corraware into one class, which we will label *isaware*; this class thus represents all turns in which users become aware of a problem. From a system perspective, such a 3-way classification would be useful in identifying the existence of a prior system failure and in further identifying those turns which simply represent corrections; such information might be as useful, potentially, as the 4-way distinction, if we could achieve it with greater accuracy.

Indeed, when we predict the three classes (isaware, corr, and norm) instead of four, we do improve in predictive power — from 74.23% to 81.14% (+/– 0.83%) classification success. Again, this compares to the baseline (predicting norm, which is still the majority class) of 57%. We also get a corresponding improvement in terms of precision and recall, as shown in Table 5, with the isaware category considerably better distinguished than either aware or corraware in Table 3. The ruleset for the 3-class predictions is given in

	Precision (%)	Recall(%)
norm	84.49	87.48
corr	72.07	67.38
isaware	80.52	77.07
Accuracy: 81.14% (± 0.83%); baseline: 57%		

Table 5: 3-way Classification Performance.

Figure 4. The distribution of features in this ruleset is quite similar to that in Figure 3. However, there appear to be clear differences in which features best predict which classes. First, the features used to predict corrections are balanced between those from the current turn and features from the preceding context, whereas isaware rules primarily make use of features of the preceding context. Second, the features appearing most often in the rules predicting corrections are durational features (dur2, predur1, dur), while duration is used only

if (gram=universal) ∧ (dur2 ≥ 7.31) **then** *CORR*
if (dur2 ≥ 2.25) ∧ (priornofeatpct ≥ 0.11) ∧ (%sil ≤ 0.55) ∧ (wordsstr ≥ 4) **then** *CORR*
if (dur2 ≥ 2.75) ∧ (gram=universal) ∧ (pre%sil1 ≥ 1.17) **then** *CORR*
if (predur1 ≤ 0.24) ∧ (dur ≥ 0.85) ∧ (priornofeatnum ≥ 2) ∧ (pmnconf ≥ -3.11) ∧ (pmn%sil ≤ 0.45) **then** *CORR*
if (predur1 ≤ 0.19) ∧ (dur ≥ 1.21) ∧ (pmnf0mean2 ≥ 0.99) ∧ (predur2 ≤ 0.90) ∧ (%sil ≤ 0.70) ∧ (tempo ≤ 3.25) **then** *CORR*
if (predur1 ≤ 0.20) ∧ (ynstr=F) ∧ (pregram=cityname) ∧ (ppref0mean ≥ 171.58) **then** *CORR*
if (dur2 ≥ 0.75) ∧ (gram=cityname) ∧ (pmnsyls ≥ 3.67) ∧ (pmnconf ≥ -3.23) ∧ (%sil ≥ 0.41) **then** *CORR*
if (prenofeat=T) ∧ (preconf ≥ -0.72) **then** *CORR*
if (preconf ≤ -4.07) **then** *ISAWARE*
if (preconf ≤ -2.76) ∧ (pmntempo ≤ 2.39) ∧ (tempo2 ≥ 1.56) ∧ (preynstr=F) **then** *ISAWARE*
if (preconf ≤ -2.75) ∧ (ppau ≥ 0.46) ∧ (tempo ≤ 1.20) **then** *ISAWARE*
if (pretempo ≤ 0.23) **then** *ISAWARE*
if (pmnconf ≤ -3.10) ∧ (ppregram=universal) ∧ (ppre%sil ≤ 0.34) ∧ (tempo1 ≤ 2.94) **then** *ISAWARE*
if (predur ≥ 1.27) ∧ (pretempo ≤ 2.36) ∧ (prermsmean ≥ 229.33) ∧ (tempo2 ≥ 0.83) **then** *ISAWARE*
if (preconf ≤ -2.80) ∧ (nofeat=T) ∧ (f0mean ≤ 205.56) **then** *ISAWARE*
else *NORM*

Figure 4: Rules for Predicting 3 Turn Categories.

once in isaware rules. Instead, these rules make considerable use of the ASR confidence score of the preceding turn; in cases where aware turns immediately follow a rejection or recognition error, one would expect this to be true. Isaware rules also appear distinct from correction rules in that they make frequent use of the tempo feature. It is also interesting to note that rules for predicting isaware turns make only limited use of the nofeat feature, i.e. whether or not a variant of the word *no* appears in the turn. We might expect this lexical item to be a more useful predictor, since in the explicit confirmation condition, users should become aware of errors while responding to a request for confirmation.

Note that corrections, now the minority class, are more poorly distinguished than other classes in our 3-way classification task (Table 5). In a third set of experiments, we merged corrections with normal turns to form a 2-way distinction over all between aware turns and all others. Thus, we only distinguish turns in which a user first becomes aware of an ASR failure (our original isaware and corraware categories) from those that are not (our original corr and norm categories). Such a distinction could be useful in flagging a prior system problem, even though it fails to target the material intended to correct that problem. For this new 2-way distinction, we obtain a higher degree of classification accuracy than for the 3-way classification — 87.80% (+/– 0.61%) compared to 81.14%. Note, however, that the baseline (predict majority class of !isaware) for this new classification is 70%, considerably higher than the previous baseline. Table 6 shows the improvement in terms of accuracy, precision, and recall.

	Precision (%)	Recall (%)
!isaware	91.7	91.6
isaware	80.7	81.1
Accuracy: 87.80% (± 0.61%); baseline: 70%		

Table 6: 2-way Classification Performance.

The ruleset for the 2-way distinction is shown in Figure 5. The features appearing most frequently

if (preconf ≤ -4.06) ∧ (pretempo ≤ 2.65) ∧ (ppau ≥ 0.25) **then** *T*
if (preconf ≤ -3.59) ∧ (prerejbool=T) **then** *T*
if (preconf ≤ -2.85) ∧ (predur ≥ 1.039) ∧ (tempo2 ≥ 1.04) ∧ (preppau ≥ 0.57) ∧ (pretempo ≤ 2.18) **then** *T*
if (preconf ≤ -3.78) ∧ (pmnsyls ≥ 4.04) **then** *T*
if (preconf ≤ -2.75) ∧ (prestr contains the word 'help') **then** *T*
if (pregram=universal) ∧ (pprewordsstr ≥ 2) **then** *T*
if (preconf ≤ -2.60) ∧ (predur ≥ 1.04) ∧ (%sil1 ≤ 1.06) ∧ (prermsmean ≥ 370.65) **then** *T*
if (pretempo ≤ 0.13) **then** *T*
if (predur ≥ 1.27) ∧ (pretempo ≤ 2.36) ∧ (prermsmean ≥ 245.36) **then** *T*
if (pretempo ≤ 0.80) ∧ (pmntempo ≤ 1.75) ∧ (ppretempo2 ≤ 1.39) **then** *T*
then *F*

Figure 5: Rules for Predicting 2 Turn Categories: ISAWARE (T) versus the rest (F).

in these rules are similar to those in the previous two rulesets in some ways, but quite different in others. Like the rules in Figures 3 and 4, they appear independent of system characteristics. Also, of the contextual features appearing in the rules, about half are prosodic features and half ASR-related; and, of the current turn features, prosodic features dominate. And timing features again (especially tempo) dominate the prosodic features that appear in the rules. However, in contrast to previous classification rulesets, very few features of the current turn appear in the rules at all. So, it would seem that, for the broader classification

task, contextual features are far more important than for the more fine-grained distinctions.

5 Conclusion

Continuing our earlier research into the use of prosodic information to identify system misrecognitions and user corrections in a SDS, we have studied aware sites, turns in which a user first notices a system error. We find first that these sites have prosodic properties which distinguish them from other turns, such as corrections and normal turns. Subsequent machine learning experiments distinguishing aware sites from corrections and from normal turns show that aware sites can be classified as such automatically, with a considerable degree of accuracy. In particular, in a 2-way classification of aware sites vs. all other turns we achieve an estimated success rate of 87.8%. Such classification, we believe, will be especially useful in error-handling for SDS. We have previously shown that misrecognitions can be classified with considerable accuracy, using prosodic and other automatically available features. With our new success in identifying aware sites, we acquire another potentially powerful indicator of prior error. Using these two indicators together, we hope to target system errors considerably more accurately than current SDS can do and to hypothesize likely locations of user attempts to correct these errors. Our future research will focus upon combining these sources of information identifying system errors and user corrections, and investigating strategies to make use of this information, including changes in dialogue strategy (e.g. from user or mixed initiative to system initiative after errors) and the use of specially trained acoustic models to better recognize corrections.

References

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

H. H. Clark and D. Wilkes-Gibbs. 1986. Referring as a collaborative process. *Cognition*, 22:1–39.

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

J. Hirschberg, D. Litman, and M. Swerts. 2000. Generalizing prosodic prediction of speech recognition errors. In *Proceedings of the Sixth International Conference on Spoken Language Processing*, Beijing.

J. Hirschberg, D. Litman, and M. Swerts. 2001. Identifying user corrections automatically in spoken dialogue systems. In *Proceedings of NAACL-2001*, Pittsburgh.

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

G. Levow. 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.

D. Litman, J. Hirschberg, and M. Swerts. 2000. Predicting automatic speech recognition performance using prosodic cues. In *Proceedings of NAACL-00*, Seattle, May.

S. L. Oviatt, G. Levow, M. MacEarchern, and K. Kuhn. 1996. Modeling hyperarticulate speech during human-computer error resolution. In *Proceedings of ICSLP-96*, pages 801–804, Philadelphia.

A. Shimojima, K. Katagiri, H. Koiso, and M. Swerts. 1999. An experimental study on the informational and grounding functions of prosodic features of Japanese echoic responses. In *Proceedings of the ESCA Workshop on Dialogue and Prosody*, pages 187–192, Veldhoven.

M. Swerts, D. Litman, and J. Hirschberg. 2000. Corrections in spoken dialogue systems. In *Proceedings of the Sixth International Conference on Spoken Language Processing*, Beijing.

E. Wade, E. E. Shriberg, and P. J. Price. 1992. User behaviors affecting speech recognition. In *Proceedings of ICSLP-92*, volume 2, pages 995–998, Banff.