Data-driven Models for Timing Feedback Responses in a Map Task Dialogue System

Raveesh Meena, Gabriel Skantze and Joakim Gustafson

Royal Institute of Technology (KTH)
Department of Speech, Music and Hearing
Lindstedtsvägen 24, 10044, Stockholm, Sweden

raveesh@csc.kth.se, gabriel@speech.kth.se and jocke@speech.kth.se

Corresponding author: Raveesh Meena
raveesh@csc.kth.se, +46-(0)-8-790 7872

The first draft submitted to Computer Speech and Language on 08 Oct, 2013

Abstract

Traditional dialogue systems use a fixed silence threshold to detect the end of users' turns. Such a simplistic model can result in system behaviour that is both interruptive and unresponsive, which in turn affects user experience. Various studies have observed that human interlocutors take cues from speaker behaviour, such as intonation, paralanguage, content, and gestures among others, to coordinate smooth exchange of speaking turns. However, hardly any effort has been made towards implementing these models in dialogue systems and verifying how well they model the turn-taking behaviour in human–computer interactions. In this paper, we present a data-driven approach to building models for online detection of suitable feedback response locations in the user's speech. We first collected human–computer interaction data using a spoken dialogue system that can perform the Map Task with users (albeit using a trick). Using this data we trained various models that use automatically extractable prosodic, contextual and lexico-syntactic features for online detection of feedback response locations. Next, we implemented a trained model in the same dialogue system and evaluated it in interactions with users. The results of a perception test of the user interactions support our hypothesis that the trained model provides for smoother dialogue in contrast to a baseline model. We also found that the trained model enhances the system's responsiveness in contrast to the baseline model. To our knowledge, this is the first work on actual verification of the proposals on using speaker behavioural cues such as prosody, syntax and context, in modelling human-like turn-taking behaviour in spoken dialogue systems. Our results confirm that a model trained on these speaker modalities offers both smooth turn-transitions and responsive system behaviour.

Keywords: Spoken dialogue systems; Timing Feedback; Turn-taking; User evaluation

1 Introduction

Traditionally, dialogue systems have rested on a very simple model for turn-taking, where the system uses a fixed silence threshold to detect the end of the user's utterance, after which the system responds. However, this model does not capture human–human dialogue very accurately; sometimes a speaker just hesitates and no turn-change is intended, sometimes the turn changes after barely any silence (Sacks et al., 1974). Therefore, such a simplistic model can result in systems that frequently produces responses at inappropriate occasions, or produces delayed or no response at all when expected, thereby causing the system to be as perceived as interruptive or unresponsive. Related to the problem of turn-taking is that of *backchannels* (Yngve, 1970). Backchannel feedback – short acknowledgements such as *uh-huh* or *mm-hm* – are used by human interlocutors to signal continued attention to the speaker, without claiming the conversational floor. If a dialogue system should be able to manage smooth exchange of speaking turns and provide backchannel feedback without being interruptive, it must be able to first identify suitable locations in the user's speech to do so.

Human conversational partners are skilled at managing smooth turn-transitions. Duncan (1972) observed that human interlocutors continuously monitor cues, such as content, syntax, intonation, paralanguage, and body motion, in parallel to manage turn-taking. Similar observations have been made in various other studies investigating the turn-taking and back-channelling phenomena in human conversations. Ward (1996) has suggested that a low pitch region is a good cue that backchannel feedback is appropriate. On the other hand, Koiso et al. (1998) have argued that both syntactic and prosodic features make significant contributions in identifying turn-taking and back-channelling relevant places. Cathcart et al. (2003) have shown that syntax in combination with pause duration is a strong predictor for backchannel *continuers*. Gravano & Hirschberg (2011) identified seven turn-yielding and six backchannel-yielding cues spanning over prosodic, acoustic, and lexico-syntactic feature that could be used for recognition and generation of turns and backchannel.

However, there is a general lack of studies on how such models could be used online in dialogue systems and to what extent that would improve the interaction. There are two problems in doing so. First, the data used in the studies mentioned above are from human–human dialogue and it is not obvious to what extent the models derived from such data transfers to human–machine dialogue. Second, many of the features used in the proposed models were manually extracted. This is especially true for the transcription of utterances, but several studies also rely on manually annotated prosodic features.

In this paper, we present a data-driven model of what we call *Response Location Detection* (RLD), which is fully online. Thus, it only relies on automatically extractable features—covering syntax, prosody and context. The model has been trained on human–machine dialogue data and has been implemented in a dialogue system that is in turn evaluated with users. The setting is that of a Map Task, where the user describes the route and the system may respond with, for example, acknowledgements and clarification requests. The presented approach exemplifies a boot-strapping procedure where more and more advanced versions of the system are built iteratively. After each iteration, users interact with the system and data is collected. This data is then used to improve the data-driven models in the system.

In section 2 we discuss previous studies on cues that human interlocutors use to manage turn-taking and backchannels. We will also discuss some of the proposed computational models. In section 3 we describe the test-bed that we used for boot-strapping a Map Task dialogue system to collect data and develop an improved incremental version of the system. In section 4 we will discuss the various data-driven models that we have explored in this work. We describe the various the features and their performance with various learning algorithms that we have tested on our data for online use. In section 5, we discuss the subjective and objective evaluation schemes used for verifying the contributions of the trained model in user interactions. We discuss the contributions and limitations of the models presented in this paper and conclude with some ideas for future extensions of this work in in section 6.

2 Background

Two influential theories that have examined the turn-taking mechanism in human conversations are the signal-based mechanism of Duncan (1972) and the rule-based mechanism proposed by Sacks et al. (1974). According to Duncan, "the turn-taking mechanism is mediated through signals composed of clear-cut behavioural cues, considered to be perceived as discrete". Duncan identified six discrete behavioural cues that a speaker may use to signal the intent to yield the turn. These behavioural cues are: (i) any deviation from the sustained intermediate pitch level; (ii) drawl on the final syllable of a terminal clause; (iii) termination of any hand gesticulation or the relaxation of tensed hand position—during a turn; (iv) a stereotyped expression with *trailing off* effect; (v) a drop in pitch and/or loudness; and (vi) completion of a grammatical clause. Speakers may display these behavioural cues either singly or together, and when displayed together they may occur either simultaneously or in tight sequence. In his analysis, Duncan found that the likelihood of listener attempts to take the turn increase in a strictly linear fashion as more yielding cues are conjointly displayed. According to the rule-based mechanism of Sacks et al. (1974) turn-taking is regulated by applying rules (e.g. "one party at a time") at Transition-Relevance Places (TRPs)—possible completion points of basic units of turns, in order to minimize gaps and overlaps. The basic units of turns (or turn-constructional units) include sentential, clausal, phrasal, and lexical constructions.

While these theories have offered a function-based account of turn-taking, another line of research has looked into corpora-based techniques to build models for detecting turn-transition and feedback relevant places in speaker utterances.

Ward (1996) suggested that a 110 millisecond (ms) region of low pitch is a fairly good predictor for back-channel feedback in casual conversational interactions. He also argued that more obvious factors, such as utterance end, rising intonation, and specific lexical items, account for less than they seem to. He contended that prosody alone is sometimes enough to tell you what to say and when to speak. Truong et al. (2010) presented an extension to this model by additionally including pause information. They observed that the length of a pause preceding a backchannel is one of the important features in their model, next to the duration of the pitch slope at the end of an utterance.

Koiso et al. (1998) analysed prosodic and syntactic cues to turn-taking and backchannels in Japanese Map Task dialogs. They observed that some part-of-speech (POS) features are strong syntactic cues for turn-change, and some others are strongly associated with no turn-change. Using manually extracted prosodic features for their analysis, they observed that falling and rising F0 patterns are related to changes of turn, and flat, flat-fall and rise-fall patterns are indications of the speaker continuing to speak. Extending their analysis to backchannels, they asserted that syntactic features, such as filled pauses, alone might be sufficient to discriminate when back-channelling is inappropriate, whereas presence of backchannels is always preceded by certain prosodic patterns.

Cathcart et al. (2003) presented a shallow model for predicting the location of backchannel *continuers* in the HCRC Map Task Corpus (Anderson et al., 1991). They explored features such as POS tag, word count in the preceding speaker utterance, and silence pause duration, in their models. A model based on silence pause only inserted a backchannel in every speaker pause longer than 900 ms and performed better than a baseline word model that predicted a backchannel every seventh word. A tri-gram POS model predicted that nouns and pronouns before a pause are the two most important cues for predicting backchannel continuers. The combination of the tri-gram POS model and pause duration model offered a five-fold improvement over the baseline model.

Gravano & Hirschberg (2011) examined seven turn-yielding cues that take place with a significantly higher frequency in speaker utterances prior to a turn transition than those preceding turn holds. These events are: (i) a falling or high-rising intonation at the end of speaker turn; (ii) an increased speaker rate; (iii) a lower intensity level; (iv) a lower pitch level; (v) a longer duration; (vi) a higher value of three voice quality features: jitter, shimmer, and noise-to-harmonic ratios; and (vii) a point of textual completion. They also showed that when several turn-yielding cues occur simultaneously, the likelihood of a subsequent turn-taking attempt by the interlocutor increase in an almost linear fashion.

Gravano & Hirschberg (2011) also investigated whether backchannel-inviting cues differ from turn-yielding cues. They examined a number of acoustic features and lexical cues in the speaker utterances preceding smooth turn-changes, backchannels, and holds. They have identified six measureable events that are strong predictors of a backchannel at the end of an *inter-pausal unit* (IPU): (i) a final rising intonation; (ii) a higher intensity level; (iii) a higher pitch level; (iv) a final POS bi-gram equal to 'DT NN', 'JJ NN', or 'NN NN'; (v) lower values of noise-to-harmonic ratios; and (vi) a longer IPU duration. They also observed that the likelihood of a backchannel increases in quadratic fashion with the number of cues conjointly displayed by the speaker.

When it comes to using these features for making turn-taking decisions in dialogue systems, there is however, very little related work. One notable exception is Raux & Eskenazi (2008) who presented an algorithm for dynamically setting *endpointing* silence thresholds based on features from discourse, semantics, prosody, timing, and speaker characteristics. The model was also applied and evaluated in the Let's Go dialogue system for bus timetable information. However, that model only predicted the endpointing threshold based on the previous interaction up to the last system utterance, it did not base the decision on the current user utterance to which the system response is to be made.

To improve current systems, we need both a better understanding of the phenomena of human interaction, better computational models and better data to build these models. As the review above indicates, a common procedure is to collect data on human-human dialogue and then train models that predict the behaviour of the interlocutors. However, we think that it might be problematic to use a corpus of human-human dialogue as a basis for implementing dialogue system components. One problem is the interactive nature of the task. If the system produces a slightly different behaviour than what was found in the original data, this would likely result in a different behaviour in the interlocutor. Another problem is that it is hard to know how well such a model would work in a dialogue system, since humans are likely to behave differently towards a system as compared to another human (even if a more human-like behaviour is being modelled). Yet another problem is that much dialogue behaviour is optional and therefore makes the actual behaviour hard to use as a gold standard. Indeed, although many of the classifiers in the studies reported above show a better performance than baseline, they typically have a fairly low accuracy or F-score. It is also possible that a lot of human behaviour that is "natural" is not necessarily preferable for a dialogue system to reproduce, depending on the purpose of the dialogue system.

A common practice for collecting realistic human–computer interaction data in the absence of a working prototype is to use a Wizard-of-Oz setup. A human wizard operates "behind the curtain" while the users are made to believe that they are interacting with a real dialogue system. While this methodology has proven to be useful, it has its limitations such as the wizard's performance may not be consistent (across users or even for the same users) (Dahlbäck et al., 1993). The responsiveness of the wizard in responding to user behaviour is another issue, which makes the method hard to use when the issue under investigation is time-critical behaviours such as turn-taking and backchannels.

An alternative to Wizard-of-Oz studies is using a "*boot-strapping*" procedure, where more and more advanced (or human-like) versions of the system are built iteratively. After every iteration, users interact with the system and data is collected. This data is then used to improve the data-driven models in the system. A problem here, however, is how to build the first iteration of the system, since many components, such as – Automatic Speech Recognition (ASR), need some data to be useful at all. In a previous study, we presented a test-bed for collecting realistic human-computer interaction – a fully automated spoken dialogue system that can perform the Map Task with a user (Skantze, 2012). By implementing a trick, the system could convincingly act as an attentive listener, without any speech recognition. The data from user interaction with the system was used to train an offline model for the task of RLD—identifying appropriate locations to give feedback. Based on automatically extractable prosodic and contextual features, 200 ms after the end of the user's speech, the trained model was able to identify response locations with a significantly higher accuracy as compared to the majority class baseline. The trained model was, however, not evaluated in user interactions.

In this paper, we extend the approach presented in Skantze (2012) in following ways: First, we use an ASR component in order to model lexico-syntactic features. Second, we explore a range of automatically extractable features for online use–covering prosody, syntax and context, and different classes of learning algorithms. We explore the contribution of each of these modalities to the task of RLD separately as well in combination. Third, we integrated a trained model in the same system used for data collection and evaluated the model online in interaction with users.

3 Bootstrapping a Map Task dialogue system

Map Task is a common experimental paradigm for studying human-human dialogue, where one subject (the information *giver*) is given the task of describing a route on a map to another subject (the information *follower*). In our case, the user acts as the giver and the system as the follower. The choice of Map Task is motivated partly because the system may allow the user to keep the initiative during the whole dialogue, and thus only produce responses that are not intended to take the initiative, most often some kind of feedback.

Implementing a Map Task dialogue system with full speech understanding would indeed be a challenging task, given the state-of-the-art in automatic recognition of conversational speech. In order to make the task feasible, we have implemented a trick: the user is presented with a map on a screen (see Figure 1) and instructed to move the mouse cursor along the route as it is being described. The user is told that this is for logging purposes, but the real reason for this is that the system tracks the mouse position and thus knows what the user is currently talking about. It is thereby possible to produce a coherent system behaviour without any speech recognition at all, only basic speech detection. This often results in a very realistic interaction, as compared to what users are typically used to when interacting with dialogue systems—in our experiments, several users first thought that there was a hidden operator behind it[1].

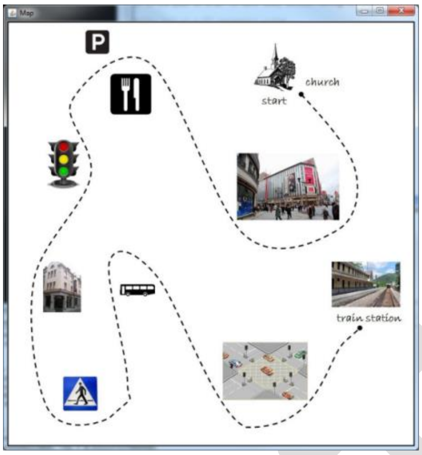

Figure 1: The user interface, of the Map Task dialogue system, showing the map.

The system is implemented using the IrisTK dialogue system framework (Skantze & Al Moubayed, 2012). The basic components of the system can be seen in Figure 2. The system uses a simple energy-based speech detector to chunk the user's speech into inter-pausal units (IPUs), that is, periods of speech that contain no sequence of silence longer than 200 ms. Such a short threshold allows the system to give backchannels (seemingly) while the user is speaking or take the turn with barely any gap. Similar to Gravano & Hirschberg (2011) and Koiso et al. (1998), we define the end of an IPU as a candidate for the Response Location Detection model to identify as a Response Location (RL). We use the term *turn* to refer to a sequence of IPUs which do not have any interlocutor responses between them.

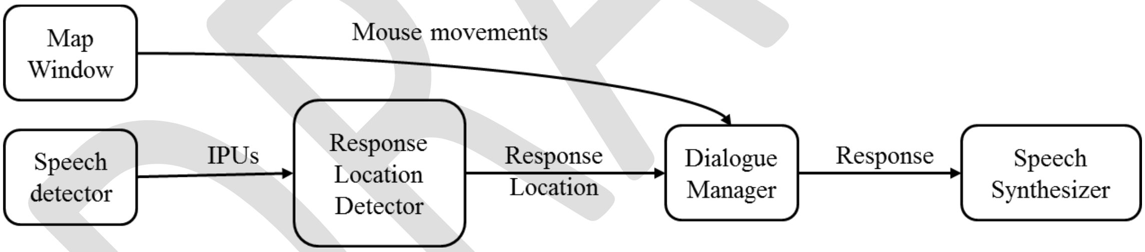

Figure 2: The basic components of the Map Task dialogue system (Iteration 1) used for data collection

Since we initially did not have any sophisticated model of RLD, it was simply set to wait for a random period between 0 and 800 ms after an IPU ended. If no new IPUs were initiated during this period, a RL was detected, resulting in random response delays between 200 and 1000 ms. Each time the RLD model detected a RL, the dialogue manager produced a Response, depending on the current state of the dialogue and the position of the mouse cursor. Table 1 shows the different types of responses the system could produce. The dialogue manager always started with an Introduction and ended with an Ending, once the mouse cursor had reached the destination. Between these, it selected from the other responses, partly randomly, but also depending on the length of the last user turn and the current mouse location. Longer turns often led to Restart or Repetition Requests, thus discouraging longer sequences of speech that did not invite the system to respond. If the system detected that the mouse had been at the same place over a longer time, it pushed the task forward by making a Guess response. We also wanted to explore other kinds of feedback than just backchannels, and therefore added short Reprise Fragments and Clarification Requests (see for example Skantze, 2007) for a discussion on these).

[1] An example video can be seen at http://www.youtube.com/watch?v=MzL-B9pVbOE.

Table 1: Different responses from the system

Introduction	"Could you help me to find my way to the train station?"
Acknowledgement	"Yeah", "Mhm", "Okay", "Uhu"
Reprise Fragment	"A station, yeah"
Clarification Request	"A station?"
Restart	"Eh, I think I lost you at the hotel, how should I continue from there?"
Repetition Request	"Sorry, could you take that again?"
Guess	"Should I continue above the church?"
Ending	"Okay, thanks a lot."

Table 2: An example Map Task dialogue between a user (U) and system(S)

S	Introduction	Could you help me to find my way to the bus stop?
U		start from the department store
S	Acknowledgement	yeah
U		and eh
S	Guess	Should I start by going west?
U		yeah do that
U		then you will get to a meadow and when you get to the meadow
S	Repeat	Eh, could you repeat that?
U		you go straight and you see a meadow on your right side
S	Clarification Request	A green field?
U		ehm yeah a field
S	Acknowledgement	mhm
U		pass the meadow and turn right so you are going north
S	Acknowledgement	okay
U		you get to a restaurant
S	Reprise Fragment	A restaurant
U		then you have to turn right and that will take you to
S	Restart	Eh, I think I lost you at the restaurant, could you take that again?
...		
U		at the junction go south and then you will get to the bus stop
S	Ending	Okay, thanks a lot.

3.1 Data collection and annotation

Ten subjects participated in the data collection. They were seated in front of the display showing the map, wearing a headset. The instructor told them that they were supposed to describe a route to the computer. They were told that they should imagine the system having a similar picture as seen on the screen, but without the route. Each subject did five consecutive tasks with five different maps, resulting in a total of 50 dialogues. Table 2 illustrates one of the example interactions with the system.

The users' speech was recorded and all events in the system were logged. Each IPU in the corpus was manually annotated into three categories: Hold (a response would be inappropriate), Respond (a response is expected) and Optional (a response would not be inappropriate, but it is perfectly fine not to respond). To validate the coding scheme two human-annotators labelled 20% of the corpus separately. For all the three categories the kappa score was 0.68, which is substantial agreement (Landis & Koch, 1977). Since only 2.1% of all the IPUs in the corpus were identified for category Optional, we excluded them from the corpus and used the data instances for the Respond and Hold categories only. The data-set contains 2272 IPUs in total; the majority of which belong to the class Respond (50.79%), which we take as our majority class baseline. Since the two annotators agreed between Respond and Hold in 87.20% of the cases, this can be regarded as an approximate upper limit for the performance expected from a model trained on this data.

In contrast to some related work (e.g. Koiso et al., 1998), we do not discriminate between locations for backchannels and turn-changes. Instead, we propose a general model for response location detection. Given the nature of the task, the system only produces utterances that are not intended to take the initiative or claim the floor, but only to provide different types of feedback (cf. Table 1). Thus, suitable response locations will be where the user invites the system to give feedback, regardless of whether the feedback is simply an acknowledgement that encourages the system to continue, or a clarification request. Moreover, it is not clear whether the acknowledgements the system produces in this domain should really be classified as backchannels, since they do not only signal continued attention, but also that some action has been performed (cf. Clark, 1996).

Another assumption behind the current model is that the system will only consider response locations at the end of IPUs. While other models have applied continuous decisions for producing backchannels (e.g., Ward, 1996), we follow the approach taken in many of the related studies mentioned above (e.g., Koiso et al., 1998; Gravano & Hirschberg, 2011). This is again partly motivated by the fact that the acknowledgements produced by the system should perhaps not be considered as backchannels. Indeed, none of the annotators felt the need to mark relevant response locations within IPUs.

4 Data-driven models for response location detection

The human–machine Map Task corpus described in the previous section was used for training a new model of RLD. We describe below how we extracted prosodic, contextual and lexico-syntactic features from the IPUs. We test the contribution of these three feature categories—individually as well as in combination, in classifying user IPUs as either Respond or Hold type. For this we explore Naïve Bayes (NB) as a generative model and three discriminative models: J48 decision tree classifier, Support Vector Machine (SVM, with radial basis kernel function) and Voted Perceptron (VP). We compare the performances of these models against the majority class baseline of 50.75% obtained by the ZeroR classifier. For all these classifiers we have used the implementations available in the WEKA toolkit (Hall et al., 2009). All results presented here are based on 10-fold cross-validation.

4.1 Prosodic features

Pitch and intensity (sampled at 10 ms) for each IPU were extracted using ESPS in Wavesurfer/Snack (Sjölander & Beskow, 2000). The values were transformed to log scale and z-normalized for each user. The final 200 ms voiced region was then identified for each IPU. For this region, the **mean pitch** and **pitch slope** (using linear regression) were used as features. We tested the impact of mean pitch in conjunction with its **absolute value**. We also explored using pitch slope in combination with its correlates, such as the **correlation coefficient *r*** for the regression line and the **absolute value of slope**. In addition to these, we also used the **duration** of the voiced region as a feature. The last 500 ms of each IPU were used to obtain the **mean intensity** and **intensity slope** measures. As with the pitch features, we tested the absolute value and the two correlates of slope for the intensity feature as well.

Table 3 illustrates the individual and collective performances of various prosodic features in classifying user IPUs as either Respond or Hold type. All pitch features combined together offer the best accuracy of 66.20% using the SVM classifier. Using all the intensity features in combination the best accuracy of 60.78% is obtained by the J48 classifier. Using all the nine prosodic features together the highest accuracy of 66.95% was obtained by the SVM classifier.

Table 3: Percentage accuracy of prosodic features in detecting response locations

		Algorithm			
#	**Feature(s)**	**J48**	**NB**	**SVM**	**VP**
1	Mean pitch	63.29	60.74	62.76	50.57
2	Mean pitch + absolute value	62.02	61.14	63.34	61.62
3	Pitch slope	62.94	59.02	57.88	55.28
4	Pitch slope + correlates	62.46	60.30	59.20	59.46
5	*All pitch features*	65.36	63.42	**66.20**	64.39
6	Mean intensity	51.98	50.31	52.20	51.36
7	Mean intensity + absolute value	50.75	50.75	52.73	51.54
8	Intensity slope	51.72	51.94	50.75	52.55

9	Intensity slope + correlates	61.27	59.11	59.73	56.78
10	*All intensity features*	**60.78**	58.49	57.17	55.24
11	Voiced region duration	56.87	58.01	55.28	53.52
12	*All prosodic features*	65.76	66.37	**66.95**	62.81

4.2 Contextual features

We have explored discourse context features such as **turn** and **IPU length** (in terms of duration in seconds) and **last system dialogue act**. We also used the **pause duration** between the onset of a speaker IPU and the end of previous speaker/system IPU. Table 4 illustrates the performance of various contextual features, individually as well as in combinations, in discriminating IPUs as Respond or Hold type. In general all features offer an improvement over the baseline accuracy of 50.75%. IPU length feature appears to generally offer slightly better performance in contrast to the turn length feature. Using all the contextual features the best accuracy is achieved by the Voted Perceptron learner using all the features 63.73%, which is significantly better than the baseline.

Table 4: Percentage accuracy of contextual features in detecting response locations

		Algorithm			
#	**Feature(s)**	**J48**	**NB**	**SVM**	**VP**
1	IPU length (in seconds)	60.87	57.39	61.22	60.92
2	Pause duration before IPU onset	57.35	53.70	56.38	54.62
3	Turn length (in seconds)	58.67	58.45	58.85	59.11
4	Last system dialogue act	54.14	54.14	54.14	53.48
5	*All features combined*	62.59	59.73	**63.73**	62.85

Dialogue act history information have been shown to be vital for predicting a listener response when the speaker has just responded to the listener's clarification request (Koiso et al., 1998; Cathcart et al., 2003; Gravano & Hirschberg, 2011; Skantze, 2012). We have observed similar rules in our Map Task corpus. One of the rules learned by the J48 decision tree classifier is: *if the last system dialogue act is Clarification or Guess (cf. Table 1), and the turn word count is less than or equal to 1, then Respond*. In other words, if the system had previously sought a clarification, and the user has responded with a yes/no utterance, then a system response is expected. A more general rule in the decision tree suggests that: *if the last system dialogue act was a Restart or Repetition Request (cf. Table 1), and if the turn word count is more than 4 then Respond otherwise Hold.* In other words, having requested the user for information the system should wait until it receives some *amount* of information from the user.

4.3 Syntactic features

As lexico-syntactic features, we used the **word form** and **part-of-speech tag** of the last two words in an IPU. All the IPUs in our Map Task corpus were manually transcribed. Filler pauses (e.g., *eh*, *uh-hun*) and other incomplete user sounds (e.g. *trai* for *train*, or *lef* for *left*) were also orthographically transcribed and assigned the class tags FP and InComp respectively. To obtain the part-of-speech tag we used the LBJ toolkit (Rizzolo & Roth, 2010). The FP and InComp tags were added to the list of 23 POS tags obtained from LBJ. Table 5 illustrates the five most frequent POS tags for the last two words in IPUs corresponding to the Respond and Hold categories. The differences in the phrase final POS tags and their respective frequencies suggest that some POS tags have strong discriminatory power.

Table 5: The five most frequent phrase final POS tags for the Respond and Hold type class

Respond				**Hold**			
IPU final phrase POS pattern	Count	Percent	Example	IPU final phrase POS pattern	Count	Percent	Example
DT NN	261	22.60%	*the church*	PRP VBP	119	10.60%	*you go*
NN NN	160	13.90%	*grass field*	<s> FP	97	8.70%	*eh*

<s> UH	81	7%	*yes*	DT NN	73	6.50%	*the garage*
VB RB	79	6.90%	*walk south*	<s> NN	58	5.30%	*hotel*
<s> NN	74	6.4%	*field*	<s> RB	57	5.10%	*south*

Table 6 illustrates the discriminatory power of various lexico-syntactic features using the six classifiers. The results under column sub-heading “Text” are accuracy scores achieved on feature values extracted from the manual transcriptions of the IPUs. Using only the last word the best accuracy of 83.98% was achieved by the SVM classifier. The addition of the second last word generally does not result in any further improvement. The best accuracy using the last two words was achieved by the Voted Perceptron classifier, 83.10%. The POS tag feature for the last two words in IPUs offer the best accuracy of 81.47% with the J48 classifier. While POS tag is a generic feature that would enable the model to generalize, using word form as a feature has the advantage that some words, such as *yeah*, are strong cues for predicting the Respond class, whereas fillers, such as *ehm*, are strong predictors of the Hold class. Using word form and POS tag features in combination doesn’t result in large improvements over using word form alone. The best accuracy corresponding to this feature combination is achieved by the Naïve Bayes learner, 83.67%.

Table 6: Percentage accuracy of lexico-syntactic features in detecting response locations

		Algorithm							
		J48		**NB**		**SVM**		**VP**	
#	**Feature(s)**	**Text**	**ASR**	**Text**	**ASR**	**Text**	**ASR**	**Text**	**ASR**
1	Last word	82.53	80.11	82.53	80.59	**83.98**	81.12	83.85	81.47
2	Last two words	82.53	80.46	82.31	80.33	81.56	78.87	**83.10**	80.81
3	Last word’s POS tag	79.49	74.87	79.49	74.87	79.58	75.00	79.45	74.91
4	Last two word’s POS tags	**81.47**	76.54	80.37	75.84	80.55	75.31	81.12	76.36
5	Last two words + POS tags	82.53	80.46	83.67	79.62	81.78	78.83	83.49	80.85
6	Last word’s Semantic tag	83.45	79.27	83.45	79.27	83.45	79.14	83.36	78.21
7	Last two word’s Semantic tags	**83.45**	79.49	81.25	77.16	82.61	79.23	83.10	79.09
8	Last two words + Semantic tags	83.76	81.87	84.15	81.16	82.53	78.96	**84.42**	81.47
9	Last two words + ASR word confidence scores	--	80.11	--	80.59	--	78.70	--	80.81
10	Last two words + POS tags + ASR word confidence scores	--	80.50	--	80.19	--	78.65	--	80.94
11	Last two words + Semantic tags + ASR word confidence scores	--	81.87	--	81.07	--	79.09	--	**82.04**

An RLD model for online predictions requires that the syntactic features are extracted from the output of a speech recogniser. Since speech recognition is prone to errors, an RLD model trained on manual transcriptions alone – suggesting perfect recognition – would not be robust when making predictions in noisy data. Therefore we train our RLD models on actual speech recognised results. To achieve this, we did an 80-20 split of the Map Task corpus into training and test sets respectively. The manual transcriptions of IPUs in the training set were used to train the language model of an off-the-shelf ASR system. The trained ASR system was then used to recognize the audio recordings of the IPUs in the test set. After performing five iterations of splitting, training and testing, we had obtained the speech recognised results for all the IPUs in the Map Task corpus. The mean Word Error Rate (WER) for the five iterations was 17.32% (*SD* = 4.45%).

In Table 6, columns with sub-heading “ASR” illustrate the performances of lexico-syntactic features corresponding to feature values extracted from the best speech recognized hypotheses for the IPUs. With the introduction of a word error rate of 17.32%, the performances of all the models using the feature word form slightly decline, as expected (cf. rows 1 and 2). However, in contrast to this decline the corresponding decline in accuracy for models that use only POS tag feature is much larger (cf. rows 3 and 4). This is because the POS tagger itself uses the left context to make POS tag predictions. With the introduction of errors in the left context, the tagger’s accuracy is affected, which in turn affects the accuracy of these RLD models. These performances are bound to decline further with increase in ASR errors. To identify the correlation between ASR WER and these RLD model performances we first obtained various ASR performances by using increasingly smaller training set in the iterative process, described just above for obtaining speech recognised utterances for the corpus. The size of training data in terms of number of sentences, number of words, vocabulary size, and the corresponding ASR WER obtained are summarized in Table 7.

Table 7: Impact of training data size on ASR WER

Percentage of sentences from training data set used for training	100%	10%	5%	2%	1%
Avg. number of sentences	1817.60	182.20	91.40	36.80	18.60
Avg. number of words	6138.40	628.80	326.60	123.20	62.00
Avg. vocabulary size	247.80	108.20	81.60	57.00	35.00
Avg. ASR WER	17.32%	24.11%	28.59%	37.94%	49.10%

The performances of the J48 classifier using the word form and POS tag features corresponding to these five WERs are illustrated in Figure 3. The results corresponding to 0 on WER axis correspond to model performance on feature values extracted from manual transcriptions. From Figure 3 we observe that while the performance of the RLD models decline with increase in ASR WER, word form as a feature offers constantly better performance in comparison to using POS tag feature only. This suggests that using context independent lexico-syntactic features would still offer better performance for an online model of RLD. We therefore also created a word class dictionary, which generalises words into domain-specific **semantic** classes in a simple way (much like a class-based n-gram model)**.** The semantic classes used in our dictionary were based on the domain-ontology used in our earlier work on automatic semantic interpretation of verbally given route descriptions (Meena et al., 2012a). For words that were not domain specific their most frequent POS tag was used as a class label. As a result, in our dictionary we had 12 classes, 4 of which were domain-specific (illustrated in Table 8 with some example words) and the remaining 8 classes were POS tags.

Table 8: Domain specific semantic class tags

Semantic class	**Example words**
Landmark	building, train, station, garage
Direction	left, right, north, south, northeast, downwards
Action	take, turn
Spatial Relation	after, from, at, before, front, around

Using the semantic tag feature for last two words, the best accuracy was achieved by the J48 classifier, 83.45% on manual transcriptions and 79.49% on ASR results. These figures are better than the corresponding performances of J48 using POS tag feature only, 79.46% and 74.87% respectively. The performances on ASR results, in row 7, in Table 6 suggest that using the semantic tag instead of POS tag (cf. row 4) generally improves the performance of the online model. This is also evident in Figure 3 where we observe that the semantic tag feature performs constantly better than POS tag feature despite increase in ASR errors. Combination of word form and semantic tag features offer the best accuracy of 84.42% on manual transcriptions using the VP algorithm and 81.87% on ASR results using the J48 classifier. The additional advantage of using a semantic tag feature over word form feature is that new examples could be easily added to the model without having to retrain it. However, a model trained in this manner is specific to a domain.

We have also explored the use of **word-level confidence scores** (ASR wConf) from the ASR module as another feature to possibly reinforce a learning algorithm's confidence in trusting the recognised words. Using the word-level confidence score in combination with word form usually offers a marginal improvement over using word form feature only (cf. column "ASR", rows 2 vs. 9, 5 vs. 10, and 8 vs. 11 in Table 6). This is also shown in the performance graph for this feature combination in Figure 3.

The best accuracy for an *offline* model of RLD using lexico-syntactic features is achieved by the Voted Perceptron classifier, 84.42%, using the features word form and semantic tag. For an *online* model of RLD, the best performance, 82.04%, is achieved again by Voted Perceptron classifier using the features word form, semantic tag and ASR word-level confidence score for the last two words in an IPU.

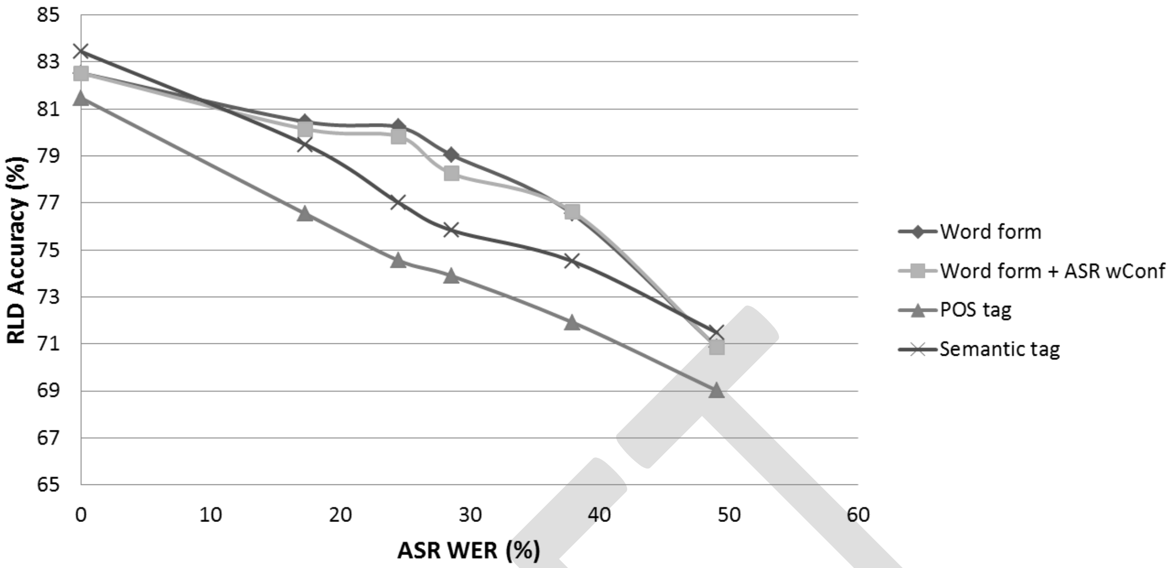

Figure 3: Performances of lexico-syntactic features using the J48 classifier, as a function of ASR WER

4.4 Combined model

Table 9 illustrates the performances of the RLD models using various feature category combinations. The top three rows are the best individual performances of the three categories. Since the prosodic and contextual features that we have used are independent of ASR WER the performances of these features categories remain unaffected despite introduction of an ASR WER of 17.32% (sub-column "ASR" in Table 9). All the model performances achieved through the combination of prosodic and contextual features exhibit improvement over using these two feature categories individually (cf. row 4, Table 9). The best accuracy for a model using context and prosody in combination is achieved by the SVM learner: 69.85%.

For the models using the lexico-syntactic features (Lex-Syntax), the figures in column with subheading "Text" are the performances excluding the word-level confidence feature, which is not relevant for manual transcriptions. Lexico-syntactic features alone provide a large improvement over not just the baseline accuracy, but also over prosody and context categories—individually as well as combined. Using prosody in combination with syntax the Naïve Bayes achieves the best accuracies, 84.64% on feature values extracted from manual transcriptions and 81.78% on values extracted from ASR results. Using prosody, syntax and context in combination, the J48 classifier achieves the best accuracies, 84.29% on manual transcriptions and 82.44% on ASR output. These figures are significantly better than the majority class baseline of 50.75% and approach the expected upper bound—the inter-annotator agreement of 87.20% on Hold and Respond types in our Map Task corpus.

Table 9: Percentage accuracy of combined models [* figures under column "Text" exclude the additional word-level confidence feature]

		J48		NB		SVM		VP	
#	Feature(s)	Text	ASR	Text	ASR	Text	ASR	Text	ASR
1	Prosody	65.76	65.76	66.37	66.37	**66.95**	**66.95**	62.81	62.81
2	Context	62.59	62.59	59.73	59.73	63.73	63.73	**62.85**	**62.85**
3	Lex-Syntax*	83.76	81.87	84.15	81.07	82.53	79.09	**84.42**	**82.04**
4	Prosody + Context	68.13	68.13	69.23	69.23	**69.85**	**69.85**	66.99	66.99
5	Prosody + Lex-Syntax*	84.11	81.69	**84.64**	**81.78**	80.5	77.51	80.85	78.39
6	Prosody + Context + Lex-Syntax*	**84.29**	**82.44**	84.24	82.00	80.72	78.21	78.96	78.21

Table 10 shows the results of significance tests on the performance of J48 classifier using some of the feature category combinations presented in Table 9. For these tests we used performance scores obtained from 10 repetitions of 10 fold cross-validation using the Weka toolkit. A Mann-Whitney U test suggests that using prosodic features alone results in significant improvement over the majority class baseline of 50.75%. We performed the Mann-Whitney U because the performance scores obtained by the baseline learner, ZeroR, were not normally distributed. For the remaining tests, a two-tailed t-test for independent samples was applied. Prosody achieves significantly better performance in contrast to using context alone. Using prosodic and contextual features in combination offers significant improvement in performance over using prosodic features alone. Syntax alone offers significant performance gain over using prosodic and context features together. Using prosody in addition to syntax doesn't offer any significant improvement over using syntactic features alone. However, when context is used as another additional feature the resulting gain in performance is significant. A Kruskal-Wallis H test indicate that performances of the five classifiers are significantly different (H(4) = 364.345, p < 0.001). A post-hoc test indicates that all learners are significantly better than the ZeroR learner (baseline), (p <0.001). Among the four learners, all learner performances except for those of SVM and Voter Perceptron (p = 0.393), and Naïve Bayes and J48 (p = 0.262), are significantly different, with p < 0.001 for the remaining pairs.

Table 10: Significance testing of feature performances on ASR results using J48 classifier (< significant difference, ≡ no significant difference)

Significance tests for various feature category comparisons
Baseline < Prosody, Mann-Whitney U test (U = 0.000, p < 0.001)
Context < Prosody (t = -7.787, df = 198, p < 0.001)
Prosody < Prosody + Context (t = -5.338, df = 198, p < 0.001)
Prosody + Context < Lex-Syntax (t = -33.141, df = 198, p < 0.001)
Lex-Syntax ≡ Prosody + Lex-Syntax (t = -0.218, df = 198, p = 0.827)
Lex-Syntax < Prosody + Context + Lex-Syntax (t = -2.628, df = 198, p = 0.009)

Table 11 illustrates the precision (fraction of correct decisions in all model decisions), recall (fraction of all relevant decisions correctly made) and F-measures for the five classifiers, using all the three feature categories together, with feature values extracted from ASR results. An ideal model would have a high precision and recall for both Respond and Hold prediction classes. The J48 classifier appears to balance this aspect, and has the highest F-measures: 84% for Respond and 81% for Hold.

Table 11: Precision (P), Recall (R) scores and F-measures (F) (in %) of the RLD model combining prosodic, contextual and syntactic features with values extracted from ASR results.

	ZeroR			**J48**			**NB**			**SVM**			**VP**		
Prediction class	**P**	**R**	**F**	**P**	**R**	**F**	**P**	**R**	**F**	**P**	**R**	**F**	**P**	**R**	**F**
Respond	51	100	67	80	88	**84**	81	84	83	75	86	80	79	78	78
Hold	00	00	0	86	77	**81**	83	80	81	83	71	76	77	79	78

4.5 RLD model performances vs. ASR performances

The performance figures in Table 9 and the t-tests suggests that the gain in model performances achieved by using prosodic and contextual features in addition to the lexico-syntactic feature is not significant, and therefore the model combination of these three feature categories may not serve any purpose. However, the role of prosodic features is emphasised when syntax alone can't disambiguate (Koiso et al., 1998) or when errors in speech recognition impair the strength of syntactic features. Figure 4 illustrates the performances of the J48 Bayes classifier corresponding to various ASR WER and using various feature category combinations. As expected, the performance of the model using only lexico-syntactic features declines (linearly, R^2 = 0.91) with an increase in WER. In contrast, the models using only prosody or only context, or prosody and context in combination achieve a rather stable performance. Thus prosody and context may provide features that are robust to noise in contrast to lexico-syntactic features. This is demonstrated by the performance curve of the model combining prosodic, context, and lexico-syntactic features. Corresponding to the WER of 49.10% the J48 classifier achieves an accuracy of 71.48% using only lexico-syntactic features, whereas in combination with the prosodic and contextual models (accuracy of 68.13%), the model achieves an accuracy of 74.78%, an absolute performance gain of 3.30%, which is significant.

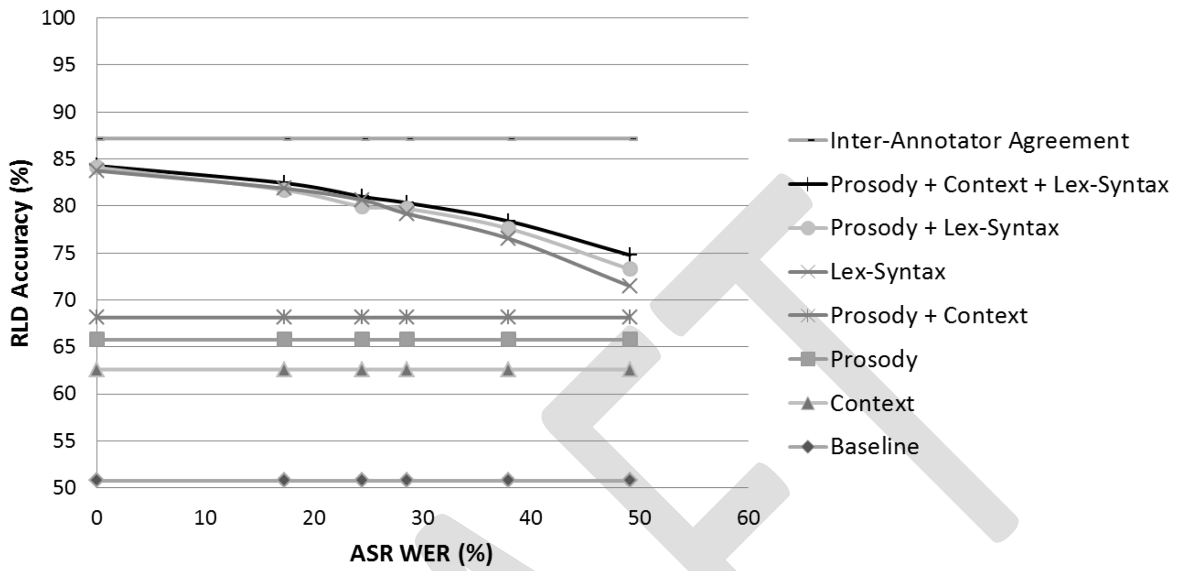

Figure 4: Performances of the J48 model of RLD as a function of ASR WER

An example illustrating the role of prosody when syntax could be misleading: The user utterance "*pass through"* was recognized as "*bus tunnel"* against the WER of 49.01%. The expected RLD decision is a Hold as the user phrase is syntactically incomplete. However, using the erroneously ASR output, which is syntactically complete, the lexico-syntactic model identified the utterance (an IPU) as Respond type. The prosodic model on the other hand classified the model as Hold type. The model combining both syntactic and prosodic features classified the IPU as a Hold type.

Due to our simplistic method for extraction of prosodic feature, at times, the prosodic features lead to incorrect decisions as well. For example, the user utterance "*then you will get to a corner where you have a church on your right,*" was recognised as "*then the way yes the before way you eh the church all and garage*" (against WER 49.01%). The expected RLD decision is to label the user IPU as Respond type. The lexico-syntactic model identified the IPU as Respond type. The prosodic model identified it as Hold type. The model using both syntactic and prosodic features, however, falsely classified the IPU as Hold type.

As regard to the contribution of context to the model combination of prosody, context, and lexico-syntactic features, context does play a role when both syntax and context can't disambiguate. As an illustration of how context helps, the user utterance "*go south until you reach the pedestrian crossing*" was expected to be classified by the model as Respond type. However, the utterance was recognised as "*go south continue the church the the the station go see*" against the WER of 49.01%. The syntactic and the prosodic models both identified the ASR recognised utterance as Hold type, the model combining syntax and prosody also identified it as Hold type, however, combining contextual features to the model resulted in classifying the utterance as a Respond type. The IPU and turn length feature contributed to this decision. Other instances where context contribute is when the system should acknowledge the users' affirmative response to a previously asked clarification question.

5 User evaluation

At a quantitative level, the best accuracy of 82.44% in discriminating user utterances as Respond and Hold type, achieved by the J48 classifier –using prosodic, contextual and lexico-syntactic features in combination and extracted from ASR results with 17.32% WER– is significantly better than the majority class baseline performance of 50.75%. Would such a trained model also be perceived as significantly better in managing smooth interactions with real users? In order to evaluate the usefulness of a trained model in real user interactions we conducted a user evaluation at the early stage of this work. Two versions of the Map Task dialogue system that was used to collect the corpus (cf. section 3) were created. One version used a Random model, which made a random choice between Respond and Hold type classes. The Random model thus approximates our majority class baseline. Another version of the system used a Trained data-driven model to make the classification decision. The Trained model used seven features in total, which included four prosodic features (mean pitch, pitch slope, pitch duration and mean intensity), two contextual features (turn word count and last system dialogue act), and two lexico-syntactic features (word form and ASR word-level confidence score). A Naïve Bayes classifier trained on these features achieved an accuracy of 84.60% on manual transcriptions (excludes the feature word-level confidence score) and 82.00% on ASR recognised results. For both models, if the model decision was a Hold, the system waited 2 seconds and then responded anyway if no more speech was detected from the user. Figure 5 illustrates the components of the dialogue system using the Trained RLD model. In addition to the components used in the system during the first iteration (for data collection, cf. Figure 2), this system had three new components: an ASR module for online extraction of lexico-syntactic features, a prosodic analysis component for online extraction of prosodic features, and a module to retrieve dialogue act history.

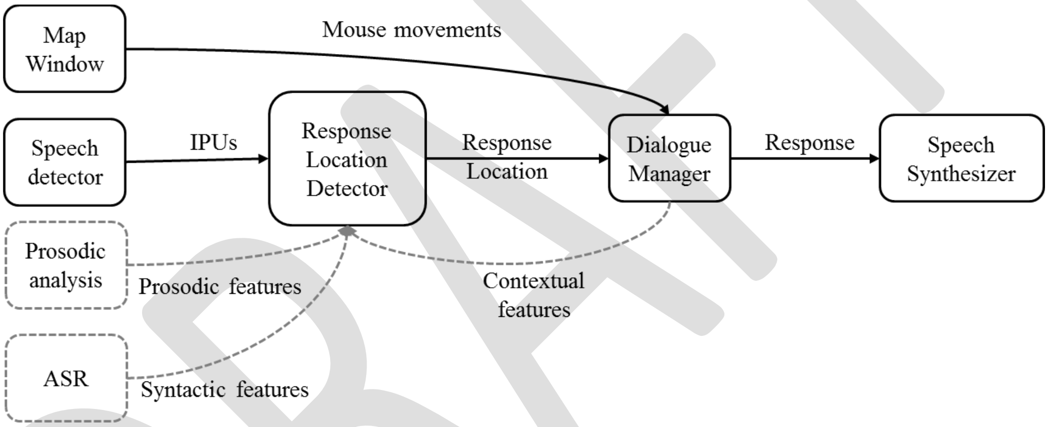

Figure 5: System architecture of the Map Task dialogue system (Iteration 2) used in user evaluation

We hypothesize that since the Random model makes random choices, it is likely to produce false-positive responses as well as false-negative responses in equal proportion. While the false-positive responses would result in occasional overlaps and interruptions in interactions, the false-negative responses would result in gaps, delayed responses or simultaneous starts during the interactions. The Trained model on the other hand would produce fewer overlaps and gaps which would provide for a smooth interaction and user experience.

In order to evaluate the two models, 8 subjects (2 female, 6 male) were asked to perform the Map Task with the two systems. Subjects were recruited from the school of Computer Science and Communication at KTH. Each subject performed five dialogues in total. This included 1 trial session with the Trained model and 2 tests each with both versions of the system. This resulted in 16 test dialogues each for the two systems. The trial session was used to allow the users to familiarize themselves with the dialogue system. Also, the audio recording of the users' speech from this session was used to normalize the user pitch and intensity for the online prosodic extraction. The order in which the systems and maps were presented to the subjects was varied over the subjects to avoid any ordering effect in the analysis.

The 32 dialogues from the user evaluation were, on average, 1.7 min long (*SD* = 0.5 min). The duration of the interactions with the Random and the Trained model were not significantly different. A total of 557 IPUs were classified by the Random model whereas the Trained model classified 544 IPUs. While the Trained model classified 57.7% of the IPUs as Respond type the Random model classified only 48.29% of the total IPUs as Respond type, suggesting that the Random model was somewhat quieter.

It turned out that it was very hard for the subjects to perform the Map Task and at the same time make a valid subjective comparison between the two versions of the system especially since we only wanted the subjects to assess the *response timing*, and not the appropriateness of the specific *response type*. Therefore, we conducted another subjective evaluation to compare the two systems. We asked subjects to listen to the Map Task user interactions and press a key whenever a system response was either lacking or inappropriate. The subjects were asked not to consider *how* the system actually responded, only evaluate the timing of the response.

5.1 Perception test

Eight users participated in the subjective judgment task. Although five of these were from the same set of users who had performed the Map Task in the user evaluation, none of them got to judge their own interactions. The judges listened to the Map Task interactions in the same order as the users had interacted, including the trial session. Whereas it had been hard for the subjects who participated in the dialogues to characterize the two versions of the system, almost all of the judges could clearly tell the two versions apart (without being told about the properties of the two versions). They stated that the Trained system provided for a smooth flow of dialogue, compared to the Random system.

A total of 149 key-press instances for the Random model and 62 key-press for the Trained model were obtained. Since the judges were asked to simply press a key, we did not have access to the information whether a key press was due to perceived inappropriate response location (false-positive model decisions) or absolute lack of a system response (false-negative model decisions). To obtain this information we analysed all the turn-transition instances where judges pressed the key. The timing of the IPUs was aligned with the timing of the judges' key-presses in order to measure the numbers of IPUs that had been given inappropriate response decisions. We found that 11 instances of key-press could be attributed to the failure of the voice-activity detector in detecting user speech immediately after a system response or during the 1.5 seconds following a Hold decision. Although judges were instructed not to judge the system utterance on the bases of the type of the response, 4 key-press instances were identified against responses that we believe were at appropriate response locations, but with inappropriate response type. There were 4 instances of key-press where the system had correctly detected a Hold, but the current position of mouse cursor on the destination landmark triggered a system response of type End (cf. Table 1), which was perceived as inappropriate by the judges. Two instances of judge key-press could not be associated with any IPU and no plausible reasons could be identified as to why the key-press occurred. We excluded these 21 instances from our analysis of the perceived inappropriateness or lack of system responses, as it would be inappropriate to hold the RLD model responsible for these decisions. A Chi-Squared one-variable test suggest that the proportion of key-presses received by the two systems (Random: 141 and Trained: 49) are significantly different ($\chi^2 = 44.54, \mathrm{df} = 1, p < .001$). We can therefore conclude that the Trained model was perceived to have significantly less inappropriate turn-transition instances than the Random model.

The perceived inappropriateness of system responses i.e., responding when it is inappropriate, could be ascribed to a false-positive (FP) decision by the RLD model, whereas the lack of response from system when it is expected could be ascribed to a false-negative (FN) decision by the RLD model. Figure 6 illustrates the distribution of the perceived FP and FN decisions made by the Random and the Trained models. We verified whether judges' key-press was biased towards false-positive or false-negative system responses. A Chi-squared test of independence of categorical variables suggests that there is insufficient evidence to conclude that the number of key-presses for one of the two models is influenced by the judges' perception of either of the FN or FP ($\chi^2 = 1.0, \mathrm{df} = 1, p = .317$). In fact both the models seem to have received key-presses for FN and FP decisions in equal proportion.

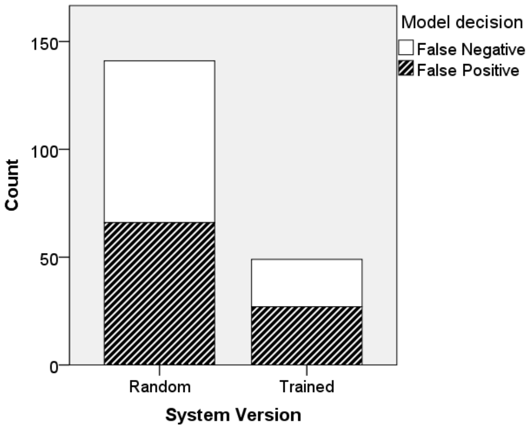

Figure 6: Overall distributions of perceived false-positive and false-negative model decisions

5.2 Response time

Responsiveness of a dialogue system has also been identified as an important issue in turn-management that affects user experience (Ward et al., 2005). Delayed system responses or lack of them could be confusing for users – they have no idea whether the system is still processing their input and they should wait for system response, or whether the system has heard them at all, which may prompt them to repeat their utterances. The repeated user utterances may cause processing overhead for the system, which could result into even longer processing and response time. As mentioned earlier, it was difficult for the users, who interacted with the Map Task system in the user evaluation, to give proper feedback regarding the responsiveness of the systems. However, the judges in the perception test were able to perceive delayed system responses. While Figure 6 suggests that both the systems have received key-presses for the FN and FP model decisions in equal proportion, we have observed that the tendencies to press a key when the response didn’t appear in expected places (or was delayed) vary across judges. Our intuition is that some judges adapted to the delay in system responses (as a consequence of timeout after an actual FN), and therefore didn’t press the key, thereby resulting in a perceived true-positive. However, it would of course be preferable if the system could have replied as quickly as possible. Therefore, an additional way of comparing the two system versions is to measure the response time of responses that were perceived as true positive.

Among the perceived true-positive system responses, a total of 267 for the Random and 312 for the Trained model, we have two categories: responses which were produced early (actual true-positive model decision) and those which were delayed (timeout after an actual false-negative model decision). The mean response time for the early responses was 301 ms ($SD = 2$ ms) whereas for the delayed responses it was 2324 ms ($SD = 4$ ms). These figures include the 200 ms silence threshold for triggering the RLD task, and the additional 2 second of wait in case of FN decisions. We can say that our system requires on an average 100 ms for processing and decision making. Table 12 shows the distribution of early and delayed responses for the Trained and the Random model. A Chi-Squared test suggests that the mean number of early and delayed responses for the two models differ significantly ($\chi^2 = 27.844, \mathrm{df} = 1, p < .001$). We can therefore conclude that the Trained system has statistically significant faster response time than the Random system.

Table 12: Distribution of early and delayed response for the Random and Trained model

Model (instances)	Early	Delayed
Random (267)	78.7%	21.3%
Trained (312)	93.6%	6.4%

5.3 Precision, Recall and F-score of the online model of RLD

An ideal data-driven model would have both high precision and recall. Table 13 shows that on the training dataset the precision, recall and F-measure of the Naïve Bayes model were significantly better in contrast to the majority class baseline model, ZeroR. To measure whether similar results were obtained during the user evaluation one would need some gold standards to compare with. One way to do this is that the annotators who labelled the training data annotate the user evaluation interaction data as well, and use them as gold standard. However, such an evaluation – at the best – would help confirm if the models' online performances were as per the annotators. An alternative – which is perhaps even stronger and viable – is to use the judges' key-press feedback from the subjective evaluation as gold standards.

Table 13: The recall, precision and F-measure of the ZeroR and the NB models in offline scenario

Prediction class	Precision (%)		Recall (%)		F-measure (%)	
	ZeroR	NB	ZeroR	NB	ZeroR	NB
Respond	50.0	81.0	100.0	87.0	66.6	83.8
Hold	0.0	85.0	0.0	78.0	0.0	81.3

In section 5.1 we used judges' key-press to identify perceived false-positive and false-negative system decisions. We considered the remaining instances of system responses as True-Positive (TP) and system holds as True-Negative (TN). Using the counts of FP, FN, TP and TN we obtained the perceived precision, recall and F-measure scores for the two models, as shown in Table 14. When compared with Table 13 these figures confirm that during the real user interactions as well the Trained model achieved better recall, precision and f-measure for both Respond and Hold type decisions, in contrast to the Random model.

Table 14: The *perceived* recall, precision and F-measure of the Random and NB models in online scenario

Prediction class	Precision (%)		Recall (%)		F-measure (%)	
	Random	NB	Random	NB	Random	NB
Respond	75.4	91.4	73.2	92.8	74.3	92.1
Hold	74.2	90.4	76.3	88.5	75.2	89.4

6 Conclusion and Discussion

We have presented a data-driven approach for RLD – detecting suitable feedback response locations in the user's speech. In contrast to the traditional procedure of using human–human interaction data to model human-like behaviour in dialogue systems, we have used actual human–computer interaction data that we collected using a fully automated dialogue system that can perform the Map Task with the user (albeit using a trick). Two annotators labelled all the user inter-pause units with regard to whether a system response is required at the end of a unit. We used automatically extractable features – covering prosody, context, and lexico-syntax – and trained various models for offline as well as *online* detection of feedback response locations. We have presented the performances of these feature categories, individually as well as in combination, for the task of RLD, using both generative and discriminative classifiers. To evaluate the contributions of such a trained model in real interactions with the user, we integrated a trained model in the same dialogue system that was used to collect training data, and tested it in user interactions. The results from the perception test of user interactions suggest that the trained model offered significantly fewer instances of inappropriate turn-transitions in contrast to a baseline model. We also found that the trained model is significantly better at being responsive in comparison to the baseline system. To our knowledge, this is the first work on actual verification of the contributions of the type of models proposed in literature for modelling human-like turn-taking and back-channelling behaviour in dialogue systems. Our results from the user evaluation suggest that a model for turn-taking trained on prosodic, contextual and lexico-syntactic features offers both smooth turn-transitions and responsive system behaviour.

In contrast to the earlier works where only the performance of offline models are discussed, we have trained and demonstrated model performances for *online* detection of response locations. We have explored the contributions of features pertaining to prosody, context, and syntax for the task at hand, whereas earlier works have used features covering only prosody (Ward, 1996), or combinations of lexico-syntax and prosody (Koiso et al., 1998), lexico-syntax and context (Cathcart et al., 2003), prosody and context (Skantze, 2012), or prosody, context, and semantics (Raux & Eskenazi, 2008). Almost none of the models proposed earlier have been tested in user interactions, with Raux & Eskenazi (2008) being an exception. However, Raux & Eskenazi (2008) excluded prosodic features from the online model used for live user interactions. They used latency as the objective measure to evaluate the model, but no subjective evaluation was carried out. We have evaluated the usefulness of our data-driven approach for RLD, by integrating a trained model in a dialogue system and testing it in user interactions. We have presented both subjective as well as objectives metrics for evaluating the improvements achieved by the trained model in contrast to a baseline model.

Using the prosodic feature values extracted by our methods, a SVM classifier could discriminate – *online* – with an accuracy of 66.95% the user IPUs in our Map Task data that required a system response from those that did not. This is substantial improvement over a majority class baseline of 50.75% in our data. Using contextual features alone provided an accuracy of 65.27 % with the Voted Perceptron classifier. While this performance is comparable to that achieved from using prosody alone, using prosody and context in combination offered the best accuracy of 69.54% using the VP classifier – a significant improvement over the individual performances of the two feature categories. Using the lexico-syntactic features alone the best accuracy of 80.04% was achieved by the VP classifier against an ASR WER of 17.32%. A model using prosodic, contextual, and lexico-syntactic features in combination achieved the best accuracy of 81.95% using the VP classifier.

The higher accuracies achieved by using the lexico-syntactic features alone corroborates with earlier observations about their significant contributions in predicting turn-transition and backchannel relevant places location (Koiso et al., 1998; Cathcart et al., 2003; Gravano & Hirschberg, 2011). While POS tag alone is a strong generic feature for making predictions in offline models its contributions in online models is reduced due to errors in speech recognition. This is because the POS tagger itself uses the left context to make predictions, and is not typically trained to handle noisy input. We have shown that using only the word form or a semantic tag, which generalises the words into domain-specific semantic classes in a simple way (much like a class-based n-gram model) offers a better and stable performance despite speech recognition errors. However, this of course results in a more domain-dependent model.

Koiso et al. (1998) have observed that prosodic features contribute almost as strongly to response location prediction as the lexico-syntactic features. We do not find such results in our data. This difference could be partly attributed to inter-speaker variation in our training data. All the users who participated in the collection of human–computer Map Task interaction data were non-native speakers of English. Also, our algorithms for extracting prosodic features are not as powerful as the manual extraction scheme used in Koiso et al. (1998). Although prosodic and contextual features do not seem to improve the performance very much when lexico-syntactic features are available, they are clearly useful when no ASR is available (best accuracy of 69.76% as compared to the baseline of 50.75%) or when ASR performance is poor. As ASR WER approached 50% the performance of lexico-syntactic features approached the performance of the model combining both prosodic and contextual features (cf. Figure 4).

One of the rules learned by the J48 decision tree classifier corroborates with the earlier observations about the role of dialogue act in modelling feedback response (Koiso et al., 1998; Cathcart et al., 2003), more specifically: producing a feedback response to acknowledge the user feedback to a previously asked system clarification request. We have observed that inclusion of additional contextual features, such as turn and IPU duration, enhances the model's discriminatory power, when syntax and prosody together cannot disambiguate.

During the user evaluation, even though the interactions with the Trained and the Random systems lasted, on an average, for 1.7 min each, the users found it difficult to recall their experience and make a subjective comparison based on only the *timing* of system responses (and not the type). One possibility for future studies could be to use the method of collecting users' expectation and experience presented in Meena et al. (2012b). In this scheme, subjects first fill a questionnaire that has dual purpose:

(i) to obtain a measure of users' expectations (on a likert scale) regarding the system behaviour(s) under investigation; and (ii) to prime users' attention to these behavioural aspects so that the users are conscious about what to evaluate. An example question is "*I expect the system to provide feedback at appropriate occasions during my speech.*" Next the users interact with one version of the system, following which they fill the same questionnaire again, however, this time they are asked to provide feedback on their experience. An example question is "*I found that the system provided feedback responses at appropriate occasions during my speech.*" The user repeats this step for the remaining system versions as well. The users' feedback from the questionnaires – on what they expect and what they actually experienced – could then be used to draw conclusions as to which system was perceived closer to their expectations.

While the users were unable to evaluate their own experience with the two systems, when they listened to the interactions of other users they were able to easily tell the two systems apart (without being told about the properties of the two versions). Thus the perception test of user interactions helped us point out instances of perceivable incorrect model decisions. The results suggest that the Trained model produced significantly fewer instances of inappropriate turn-transitions in contrast to the Random model.

Besides the issues with coordination of speaking turns, the responsiveness of a system is crucial for keeping the users engaged. We observed that in contrast to the Random model, the Trained model has a significantly large number of responses that were produced as early as possible, and significantly less number of responses that were delayed. These results bring us to the conclusion that a model for feedback response location detection trained on features covering prosody, context, and lexico-syntax offers a dialogue system with enhanced skills for smooth coordination of speaking turns with the user, and be responsive at the same time.

While the trained model has been shown to enhance the turn-management performance of the Map Task dialogue-system, there is still room for further improvements. In the current model lexico-syntactic features appear to dominate the decision making. This is evident from the respective performances of these feature categories. Also, during user evaluation, some users commented that the system appears to respond well when they mention a landmark that is on the map (or that the system only understands words). Our observation is that if the model could better exploit prosodic cues present in users' speech, it could enhance the system's responsiveness to other subtle, yet general behavioural cues present in users' speech. In this work we have mainly used pitch and intensity related prosodic features that we could extract automatically for online use. Also, the algorithms used for extracting these feature values are very simple, and perhaps using better extraction methods would improve the model's performance. Other prosodic features such as intonation patterns, speaking rate, and acoustic cues such as jitter, shimmer and noise to harmonic ratio have been also identified as useful behavioural cues for prediction of turn-taking and backchannel relevant places (Koiso et al., 1998; Gravano & Hirschberg, 2011). Training models on these features should result in tangible and perceivable improvements in RLD model's performance. As tools for online extraction of these cues become available they could be easily incorporated in the current models. Thus, the dataset from this experiment could serve as a test for evaluating the applied usefulness of new models for prosodic analysis.

A general limitation of data-driven approaches is sparseness in training data. Due to the limited number of tasks in the Map Task user interactions, it is plausible that users' share a common lexical space; however, due to inter-speaker variations in the prosodic realizations and usage, the prosodic space is spares, which makes it difficult for the models to generalize across users. It would be interesting to explore algorithms that are better at generalizing across speakers.

6.1 Directions for future work

We have so far explored prosodic, contextual, and lexico-syntactic features for predicting response location. An immediate extension to our model would be to bring more general syntactic features in the model. The motivation for this is syntactically incomplete user utterances such as "*and after the hotel…*" Our current model that uses lexico-syntactic features only, on observing the "DT NN" phrase final pattern, would predict a Respond. However, syntactic knowledge suggests that the predicate is missing in the user utterance, and therefore the system should predict a Hold.

In a future version of the system, we do not only want to determine *when* to give responses but also *what* to respond. This would require processing at a higher level, more specifically understanding the semantics of spoken route descriptions, to play an active role in decision making. In Meena et al. (2012a) we have presented a data-driven method for automatic semantic interpretation of verbal route descriptions into *conceptual route graphs* (CRG)—a semantic representation that captures the semantics of the way humans structure information in route descriptions. A CRG with missing concepts or relations should suggest incompleteness, and therefore a Hold. However, the incompleteness could also be due to ASR misrecognitions, and perhaps the confidence scores from the ASR and the spoken language understanding component, could be used to identify *what* to respond, and also select between different forms of clarification requests and acknowledgements.

We would also like to also test whether our models for feedback response location detection would generalize to other domains. While both context and prosody offer features that are domain independent, POS tag would still be a more suitable lexico-syntactic feature in contrast to word form and semantic tag features, for a domain independent model of RLD.

Other possible extension is to situate the Map Task interaction in a face-to-face Map Task between a human and a robot (such as Furhat, Al Moubayed et al., 2013) and add features from speaker's gaze, which has been identified as visual cues that humans use to coordinate turn-taking in face-to-face interactions (Kendon, 1967; Duncan, 1972).

Acknowledgments

This work is supported by the Swedish research council (VR) project *Incremental processing in multimodal conversational systems* (2011-6237). We would also like to thank the participants of our user evaluation and perception tests.

References

Al Moubayed, S., Skantze, G., & Beskow, J. (2013). The Furhat Back-Projected Humanoid Head - Lip reading, Gaze and Multiparty Interaction. *International Journal of Humanoid Robotics, 10*(1).

Anderson, A., Bader, M., Bard, E., Boyle, E., Doherty, G., Garrod, S., Isard, S., Kowtko, J., McAllister, J., Miller, J., Sotillo, C., Thompson, H., & Weinert, R. (1991). The HCRC Map Task corpus. *Language and Speech, 34*(4), 351-366.

Cathcart, N., Carletta, J., & Klein, E. (2003). A shallow model of backchannel continuers in spoken dialogue. In *10th Conference of the European Chapter of the Association for Computational Linguistics*. Budapest.

Clark, H. H. (1996). *Using language*. Cambridge, UK: Cambridge University Press.

Dahlbäck, N., Jönsson, A., & Ahrenberg, L. (1993). Wizard of Oz studies – why and how. In *Proceedings from the 1993 International Workshop on Intelligent User Interfaces* (pp. 193-200).

Duncan, S. (1972). Some Signals and Rules for Taking Speaking Turns in Conversations. *Journal of Personality and Social Psychology, 23*(2), 283-292.

Gravano, A., & Hirschberg, J. (2011). Turn-taking cues in task-oriented dialogue.. *Computer Speech & Language, 25*(3), 601-634.

Hall, M., Frank, E., Holmes, G., Pfahringer, B., Reutemann, P., & Witten, I. H. (2009). The WEKA Data Mining Software: An Update. *SIGKDD Explorations, 11*(1).

Kendon, A. (1967). Some functions of gaze direction in social interaction. *Acta Psychologica, 26*, 22-63.

Koiso, H., Horiuchi, Y., Tutiya, S., Ichikawa, A., & Den, Y. (1998). An analysis of turn-taking and backchannels based on prosodic and syntactic features in Japanese Map Task dialogs. *Language and Speech, 41*, 295-321.

Landis, J., & Koch, G. (1977). The measurement of observer agreement for categorical data. *Biometrics, 33*(1), 159-174.

Meena, R., Jokinen, K., & Wilcock, G. (2012b). Integration of Gestures and Speech in Human-Robot Interaction. In *Proceedings of 3rd International Conference on Cognitive Infocommunications (CogInfoCom 2012)*. Kosice.

Meena, R., Skantze, G., & Gustafson, J. (2012a). A Data-driven Approach to Understanding Spoken Route Directions in Human-Robot Dialogue. In *Proceedings of Interspeech*. Portland, OR, US.

Raux, A., & Eskenazi, M. (2008). Optimizing endpointing thresholds using dialogue features in a spoken dialogue system. In *Proceedings of SIGdial 2008*. Columbus, OH, USA.

Rizzolo, N., & Roth, D. (2010). Learning Based Java for Rapid Development of NLP Systems. *Language Resources and Evaluation*.

Sacks, H., Schegloff, E., & Jefferson, G. (1974). A simplest systematics for the organization of turn-taking for conversation. *Language, 50*, 696-735.

Sjölander, K., & Beskow, J. (2000). WaveSurfer - an open source speech tool. In Yuan, B., Huang, T., & Tang, X. (Eds.), *Proceedings of ICSLP 2000, 6th Intl Conf on Spoken Language Processing* (pp. 464-467). Beijing.

Skantze, G., & Al Moubayed, S. (2012). IrisTK: a statechart-based toolkit for multi-party face-to-face interaction. In *Proceedings of ICMI*. Santa Monica, CA.

Skantze, G. (2007). *Error Handling in Spoken Dialogue Systems - Managing Uncertainty, Grounding and Miscommunication*. Doctoral dissertation, KTH, Department of Speech, Music and Hearing.

Skantze, G. (2012). A Testbed for Examining the Timing of Feedback using a Map Task. In *Proceedings of the Interdisciplinary Workshop on Feedback Behaviors in Dialog*. Portland, OR.

Truong, K., Poppe, R., & Heylen, D. (2010). A rule-based backchannel prediction model using pitch and pause information.. In Kobayashi, T., Hirose, K., & Nakamura, S. (Eds.), *INTERSPEECH* (pp. 3058-3061). ISCA.

Ward, N., Rivera, A., Ward, K., & Novick, D. (2005). Root causes of lost time and user stress in a simple dialog system. In *Proceedings of Interspeech 2005*. Lisbon, Portugal.

Ward, N. (1996). Using prosodic clues to decide when to produce backchannel utterances. In *Proceedings of the fourth International Conference on Spoken Language Processing* (pp. 1728-1731). Philadelphia, USA.

Yngve, V. H. (1970). On getting a word in edgewise. In *Papers from the sixth regional meeting of the Chicago Linguistic Society* (pp. 567-578). Chicago.