Automatically predicting dialogue structure using prosodic features

Helen Wright Hastie[1]
Massimo Poesio[1]
Stephen Isard[2]

[1]Human Communication Research Centre,
[2]Centre for Speech Technology Research, University of Edinburgh,
University of Edinburgh,
2 Buccleuch Place, Edinburgh EH8 9LW
http://www.hcrc.ed.ac.uk
email: {*helenw* [1], *poesio* [1]}*@cogsci.ed.ac.uk, stepheni@cstr.ed.ac.uk* [2]

Acknowledgements
The work reported here made use of tools and techniques developed together with our colleagues Paul Taylor and Simon King.

Running head:
Automatically predicting dialogue structure using prosodic features

Contents

Number of Pages: 32

Number of Tables: 17

Number of Figures: 4

Keywords: prosody, intonation, duration, dialogue acts, moves, games, discourse function, prediction, recognition.

Abstract

Spoken dialogue systems need to track dialogue structure in order to conduct sensible conversations. In previous work, we used only a shallow analysis of past dialogue in predicting the current dialogue act. Here we show that a hierarchical analysis of dialogue structure can significantly improve dialogue act recognition. Our approach is to integrate dialogue act recognition with speech recognition, seeking a best overall hypothesis for what words have been spoken and what dialogue act they represent, in the light of both the dialogue history so far and the current speech signal. A useful feature of this approach is that intonation can be used to aid dialogue act recognition by combining it with other information sources in a natural way.

1 Introduction

Dialogue act identification is an important task for a dialogue system. It is essential to know if the response to a system's question is an answer or an objection. In addition, it is important to establish the extent to which the user has established a conversational goal so that the dialogue system can update its knowledge base and continue the conversation in the appropriate manner. The goal of the work reported in this paper is to test whether using hierarchical information about dialogue structure leads to improved performance in dialogue act recognition.

As in previous work (Taylor *et al.*, 1998), we integrate dialogue act recognition with word recognition, in the sense that word probabilities are computed by language models specific to different dialogue acts, and dialogue act likelihoods take word probabilities into account. The hypotheses produced by the integrated system are of the form "Yes-no query consisting of 'Is it' " or "Reply consisting of 'It is' ", where, crucially, the hypothesised word string for the utterance viewed as a question need not be the same as the hypothesised word string for the same utterance viewed as a reply. As a result, the a priori most likely dialogue act can potentially be rejected on the basis of word recognition and the phonetically most likely word string can be rejected on the basis of dialogue act considerations. The viterbi architecture which achieves this integration is described in section 3. Previous studies have shown that a reduction of word error rate is obtainable by integrating dialogue act recognition into

their systems (Taylor *et al.*, 1998; Shriberg *et al.*, 1998).

The architecture described in section 3 also permits us to introduce intonational information in a natural way. Dialogue acts correlate not just with the words that are spoken, but with how they are spoken, in particular with prosody. For example, in our data the utterance "okay" is often realised with a rising intonation if it is a *checking* dialogue act and a falling intonation if it is an *acknowledgement*. Our architecture weights the likelihoods of dialogue act types for a given utterance according to their probability of occurrence with the observed intonation contour calculated using a statistical intonation model.

As well as modelling the word sequences and the intonation of various dialogue acts, our system uses a dialogue model that captures regularities in a sequence of dialogue acts. For example, a *query* followed by a *reply* followed by an *acknowledgement* is more likely than three replies in a row. The theory of dialogue that we adopt is derived from the theory of *Conversational Games* used to annotate the Map Task corpus (Power, 1979; Carletta *et al.*, 1997). According to this theory, conversations consist of a series of GAMES, each of which involves an INITIATING MOVE (such as an *instruction* or a *query*) followed by either a RESPONSE MOVE (such as an *acknowledgement* or a *reply*) or possibly an embedded game (e.g., a *query* may be followed by a *clarification* subdialogue).

Experiments reported in (Poesio & Mikheev, 1998) used the annotated Glasgow version of the Map Task corpus to compare the ability of two types of dialogue models to predict move types. The first type takes the hierarchical structure of Conversational Game Theory into account; the second simply models the sequence of moves ignoring game structure, as in the models used in Nagata & Morimoto (1994), Reithinger & Klesen (1997) and Taylor *et al.* (1998). Poesio and Mikheev found that having perfect knowledge of a move's position in a game and of the type of game leads to a 30% reduction in the error rate of move prediction.

The goal of the experiments described below, was to compare various dialogue models in terms of their ability to predict the move type of an utterance whose game information is automatically derived. We find that taking into ac-

count the position of an utterance in a game significantly improves the ability of the system to predict move type, even when this information has to be automatically extracted from the input. Further experiments show that game information also improves performance results on the task, previously attempted by Terry *et al.* (1994), of discriminating declarative and interrogative utterance types.

We also look at whether classifying utterances using game information provides a better correlation with observed intonation patterns. For example, a *ready* move at the start of a game may be more emphatic than one in the middle of a game. Finally, we test whether knowing the game position and type of a move gives us extra information about word sequence regularities. For instance, a *ready* move at the start of a game may contain a larger vocabulary than *ready* moves in the rest of the game, as these just tend to consist of "okay".

The structure of the paper is as follows. We first discuss the type of data used and the general architecture of our system. We then describe each of the statistical models in turn (dialogue, intonation and language models) and how game information can make these models more effective. Finally, we present move recognition results and discuss further possible developments.

2 The Data

The experiments reported here use a subset of the DCIEM Map Task corpus (Bard *et al.*, 1996). This is a corpus of spontaneous goal-directed dialogue speech collected from Canadian speakers. In the Map Task scenario, each conversation has two participants each playing the roles of *giver* and *follower*. Generally the *giver* is giving instructions and guiding the *follower* through the route on the map. Due to the different nature of the roles, each participant has a different distribution of moves. The Map Task corpus was chosen as it is readily available, easy to analyse and has a limited vocabulary and structured speaker roles. The DCIEM Map Task was chosen over the Glasgow Map Task as it is hand-labelled for intonation events in accordance with the Tilt Theory (Taylor, 2000). In addition, we could take advantage of the large body of

previous work on developing a good baseline recognition system for North American English.

The corpus we used consists of 25 dialogues, which we divided into a training set of 20 dialogues (3726 utterances) and a test set of five dialogues (1061 utterances)[1]. None of the test set speakers are in the training set. The data were hand transcribed at the word level and are divided into utterances where one utterance corresponds to one move.

As mentioned in the Introduction, the utterances are classified according to *Conversational Game Theory* (Power, 1979; Carletta *et al.*, 1997). The data are analysed in terms of the following categories:

- 12 move types
- position in game (start, middle, end)
- game type

Instruct	direct or indirect request or instruction. E.g. "Go round, ehm horizontally underneath diamond mine..."
Explain	provides information, believed to be unknown by the game initiator. E.g. "I don't have a ravine."
Align	checks that the listener's understanding aligns with that of the speaker. E.g. "Okay?"
Check	asks a question to which the speaker believes s/he already knows the answer, but isn't absolutely certain. E.g. "So going down to Indian Country?"
Query-yn	a yes-no question. E.g. "Have you got the graveyard written down?"
Query-w	asks a question containing a wh-word. E.g. "In where?"

Table 1: Initiating moves

Acknowledge	indicates acknowledgement of hearing or understanding. E.g."Okay."
Clarify	clarifies or rephrases old information. E.g. { so you want to go ... actually diagonally so you're underneath the great rock.} "diagonally down to uh horizontally underneath the great rock."
Reply-y	elicited response to query-yn, check or align, usually indicating agreement. E.g. "Okay.", "I do.".
Reply-n	elicited response to query-yn, check or align, usually indicating disagreement. E.g. "No, I don't.".
Reply-w	elicited response that is not to clarify, reply-y or reply-n. It can provide new information and is not easily categoriseable as positive or negative. E.g. { And across to?} "The pyramid.".
Ready	indicates that the previous game has just been completed and a new game is about to begin. E.g. "Okay.", "Right." {so we're down past the diamond mine?}

Table 2: Other moves; {} indicates previous or next move

The conversational game analysis described in Carletta *et al.* (1997) uses six games: *Instructing, Checking, Query-YN, Query-W, Explaining* and *Aligning*. The initiating moves of these games are described in Table 1, and other possible moves in Table 2. Initiating moves tend to have a higher proportion of content words than non-initiating moves, which mostly express acknowledgements and responses. In a dialogue system, it is more important to recognise the words correctly in initiating moves; the information contained in non-initiating moves is often conveyed by the move type itself. Our system is better at recognising the words of initiating moves but has a higher move type recognition accuracy for the non-initiating move set. A dialogue system would not need to distinguish between the words "yep, yes, yeah", for example, as long as it knows that the utterance is a positive reply.

For this study, game position is allocated to each utterance depending on

Speaker	Utterance	Move	Position	Game
Giver:	Mike, do you see the start?	align	start	align
Follower:	Yes I do.	reply-y	end	align
Giver:	Do you have a telephone booth just below the start?	query-yn	start	query-yn
Follower:	Yes I do.	reply-y	middle	query-yn
Giver:	Okay.	acknowledge	end	query-yn
Giver:	Go approximately one inch to the left of the telephone booth.	instruct	start	instruct
Follower:	Yes.	acknowledge	middle	instruct

Table 3: Data extract including game, position and move type

whether it is at the *start, middle* or *end* of a game. We did consider an additional label *start_end* for games containing a single move, e.g., an *align* game that contains just an *align* move between an *instruct* and a *check* game. However, initial experiments using a bigram dialogue model on transcribed data showed that including this position type did not improve recognition results. It was therefore discarded and all such moves were labelled as *start*.

Table 3 shows an extract from a dialogue annotated with move, game type and position labels. Every utterance is assigned a value for each of these three categories.

3 System Architecture

As discussed in the introduction, our system performs move recognition [2] using three types of statistical models: intonation model (IM), dialogue models (DM) and language models (LM) in addition to the output of the speech recogniser.

Although there is a correlation between intonation contour types and move types, there is not a unique mapping, any more than there is for syntactic types or dialogue contexts. For example, *align* move types are realised with both rising and falling boundary tones, possibly reflecting the level of the speaker's confidence. Wright (1999) describes methods for training stochastic models that assign a likelihood for each move type given the current pitch contour. These likelihoods are combined with the outputs of the other components to produce an overall best guess. The use of stochastic models is only successful

if each move type has a different distribution of intonation features.

In Taylor *et al.* (1998), a separate language model is trained for each move type, resulting in twelve language models for the original move set. The speech recogniser is effectively run several times in parallel, using each of the move specific LMs. Language model prior probabilities are combined with word recognition probabilities to produce likelihoods for word strings according to the various language models. For example, if the recogniser assigns a high probability to a hypothesis of the word "yes" spanning the whole utterance, the *reply-y* LM will produce a high score overall, because the probability of "yes" as a *reply-y* is also high. However, the *reply-n* LM will produce a lower score because the high recognition score for "yes" will be multiplied by a low probability of occurrence in that LM.

Finally, regularities about move types are captured by a statistical dialogue model. The dialogue models we tested use dialogue information such as the previous move type, the position of a move in a game, the type of a game, and the identities of the speakers to predict the current move type.

A viterbi search finds the most likely sequence of moves together with the words used to perform them. This process searches through all the possible move sequences, given the likelihoods from the intonation models and the language models. The probability of a sequence of moves is the product of the *transition probability*, given by the dialogue model, and the *state probability*, which is a combination of the likelihoods from the prosodic and language models. These likelihoods are weighted and summed in the log domain using the following equation, where $M\star$ is the most likely move type sequence:

$$M^{\star} \;=\; \underset{M}{\operatorname{argmax}} \left\{ w_D L^D \;+\; \sum_{i=1}^{N_U} \left(w_S L_i^S \;+\; w_I L_i^I \right) \right\} \qquad (1)$$

where L^D is the log likelihood from the dialogue model; L_i^S and L_i^I are the log likelihoods for utterance i from the speech recogniser and intonation model respectively; w_D, w_S and w_I are the weights for the three terms. This method is illustrated in figure 1, taken from Taylor *et al.* (1998).

The weights are found using a held out data set, as proposed by King

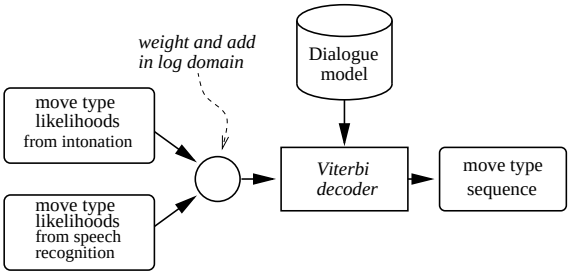

Figure 1: Finding the best move sequence

(1998). The intonation model and recogniser weights are systematically varied, while keeping the dialogue model at a fixed weight, until the optimal move recognition rate is achieved.

The result of the viterbi search is a sequence of the most likely move types for each utterance, together with the word sequence recognised by the most likely move type-specific language model. This word sequence is not irrevocably chosen before intonation and dialogue information are taken into account.

The results we present in this paper show that an improvement over previous attempts at move recognition (Taylor *et al.*, 1998) is achieved by automatically recognising the position of the utterance in a game as well as the move type. For comparison purposes, the accuracy of the system is calculated in terms of the percentage of utterances whose move type is correctly classified.

For evaluating our language, dialogue and intonation models on their own, in isolation from the rest of the system, we use the "value added" type measure of perplexity reduction. Perplexity is an information theoretic measure which rates a task as being "as hard as choosing among n equally likely possibilities" (see Rabiner and Juang (1994), p 449). The contribution that we are hoping for from each of our models is to reduce the perplexity of move recognition by as much as possible.

4 Using Dialogue Structure for Predicting Move Types

4.1 Previous Work

Poesio and Mikheev (1998) compared three dialogue models: a first one (DM 1) in which only the previous move is used for predicting the label of the

Dialogue Model	**Accuracy**
DM 1	39%
DM 1 + speaker change	42.6%
DM 1 + speaker change & speaker role	46.6%
DM 2	48.5%
DM 2 + speaker change	54.74%
DM 2 + speaker change & speaker role	58.23%
DM 3	53.8%
DM 3 + speaker change	59%
DM 3 + speaker change & speaker role	61.6%

Table 4: Results for move recognition by Poesio and Mikeev (1998)

current utterance[3]; a second one (DM 2) in which both the previous move and the position in the game of the current utterance are used as predictors; and a third one (DM 3) which is similar to the second but in which the type of game is also considered. They used the annotated Glasgow Map Task corpus to train models of move label prediction according to each of these dialogue models, using Maximum Entropy Estimation (ME) (Berger *et al.*, 1996). They found that using game position and game type improves the accuracy of move recognition from 39% (previous move only) to 48.5% for the model in which the position is also used, and 53.8% when game type is used. Adding the role of the speaker (follower or giver) and whether or not there had been a speaker change increases the accuracy from 46.6% for the basic model to 61.6% for the model using game position and game type as well. The results of these experiments are summarised in table 4.

In the experiments by Poesio and Mikheev, hand-labelled game information was used to predict move types. The work presented here attempts to provide a totally automatic system that does not rely on hand-labelled data during the test procedure.

4.2 Predicting Game Information and Move Type Separately

There are two approaches to using automatically predicted game information for move prediction. One method is to predict game information and move type separately. The second method is to predict move and game information simultaneously.

In our initial experiments we attempted to recognise game position and game type independently from move type using methods similar to those described in section 3. Specifically, intonation, language and dialogue models were trained to recognise game position and/or game type. The problem we observed with this approach is that the information we could extract automatically would not predict game type and position with a high enough degree of accuracy to lead to improvements in move recognition.

These initial dialogue modelling experiments did show that game types follow each other with a degree of regularity. For example, *align*, *check* and *explain* games are likely to be followed by an *instruct* game. These games are used to establish or check information before giving an instruction. *Query-w* and *query-yn* games, on the other hand, are typically followed by *explain* or *instruct* games. If the answer to the query is unsatisfactory, then typically an *explain* game occurs; otherwise the dialogue continues with an *instruct* game. However, game type recognition results using this method were poor. As the dialogue model can only use its own predictions, it tends to predict the same sequence repeatedly (e.g., *instruct, query-yn, explain, instruct, query-yn, explain*, etc.).

Training intonation models on game type alone would assume that utterances have similar intonation contours if they are in the same game. This is clearly wrong as a *query-yn* move in a *query-yn* game does not usually have the same intonation pattern as a *reply* in the same game type. Similarly, utterances of different move type in games of the same type would have very different wording, resulting in poor language models.

4.3 Predicting Move and Game Information Simultaneously

The second approach we tried involves creating a set of labels for utterances that encode both move type and game position and/or game type. Three methods for encoding such complex *utterance types* are given below:

- move_position (e.g., *align_middle*)
- move_game type (e.g., *align_instruct*)

Utterance type scheme	**# of types**	**Most frequent type**	**Baseline%**
position	3	*middle*	43
move	12	*acknowledge*	24
game	8	*instruct*	35
move_position	31	*acknowledge_end*	13
move_game	63	*instruct_instruct*	19
move_pos_game	117	*instruct_middle_instruct*	12

Table 5: DCIEM Map Task data statistics for training

- move_position_game type (e.g., *align_middle_instruct*)

These utterance types can be automatically recognised using the techniques for utterance classification described in section 3; the move type classification for a given utterance can then be recovered from the more complex utterance type.

Table 5 summarises the different methods of classifying utterance types in the DCIEM Map Task corpus, specifying the number of different types and the most frequent type. The baseline figure given is the percentage of utterances that would be correctly classified if the most frequent move type was picked 100% of the time.

The most common move type is *acknowledge* and the most common game type is *instruct*. Approximately 43% of the moves occur in the middle position; this is due to a certain extent to the high number of *instruct* games that have an average of four moves, i.e. an average of two middle moves per game.

One problem encountered when attempting to recognise utterance types classified using the move_game and move_position_game labelling scheme is that due to the large number of categories (63 and 117 respectively), some categories are represented by too few examples for accurate statistical modelling. In addition, if one can predict the move and position of an utterance with a degree of accuracy then the game type can be inferred from the first initiating move in the game.

A further complication is that games can be nested. In preliminary experiments, we made a distinction between nested and non-nested games. These experiments resulted in a large number of move types, creating sparse data

Move	**Start**	**Middle**	**End**	**Total**
acknowledge	0	409	510	919
align	95	22	4	121
check	185	51	9	245
clarify	0	66	25	91
explain	192	96	43	331
instruct	192	381	43	606
query-w	78	17	2	97
query-yn	237	93	4	334
ready	271	70	5	346
reply-n	0	82	25	107
reply-w	0	116	29	145
reply-y	0	201	183	384
total	1240	1604	882	3726

Table 6: Move frequencies with respect to game position

problems. In addition, whether a move is in a nested game or not may not necessarily provide useful information. Chu-Carroll (1998) ran experiments using a dialogue model that only looked at previous dialogue acts at the same level of embeddedness. That is to say, the model would only use previous utterances in the same game. Chu-Carroll shows that using this dialogue model does not result in an increase in utterance type recognition over the dialogue model that just looks at the previous utterances regardless of whether they are in the same game or not.

Given the above discussion, we concentrated on the simultaneous prediction of moves and their position in a game. The following sections discuss the results obtained by classifying utterances in terms of both their move type and game position.

5 Using Game Position in Dialogue Modelling

In studies such as Nagata & Morimoto (1994), Reithinger & Klesen (1997), Taylor *et al.* (1998), Shriberg *et al.* (1998), and King (1998), dialogue is assumed to have a flat structure, and the current dialogue act or move type is predicted on the basis of the previous utterance type only (possibly taking into account information about the current and previous speaker as well). In this section, we show that dialogue models that encode information about the

Predictor	**Symbol**
Move_position type of current move	m_p_i
Identity of speaker of current move	s_i
Identity of speaker of previous move	s_{i-1}
Move_position of previous utterance	m_p_{i-1}
Move_position of other speaker's previous utterance	m_p_{other}

Table 7: Notation of N-gram predictors

Model	**Predictors**	**Perplexity**
A	unigram	18.7
B	m_p_{i-1}	9.8
C	m_p_{i-1}, s_i, s_{i-1}	8.55
D	m_p_{other}, s_i, s_{i-1}	7.6

Table 8: Perplexity results for the different dialogue models predicting move_position categories

position of a move in a game can reduce the perplexity of the test set.

In order for this dialogue model to give good results, there must be a distinctive distribution of move types with respect to their game position. Table 6 gives the frequencies of the different moves in different game positions for the training set. From this table, one can see that there are clear patterns of move distributions across game positions. These regularities should be picked up by the dialogue model. For example, an obvious pattern is that initiating moves, with the exception of *instruct*, occur most frequently at the start of games. Most *ready* moves are game initial. Replies are quite evenly distributed across middle and end positions. All replies, with the exception of *acknowledge*, have a higher frequency of middle moves than game final moves.

Table 7 gives the types of predictors we used in training N-grams (Jelinek & Mercer, 1980) for dialogue modelling. Several combinations of these predictors were used for determining the move_position of an utterance (m_p_i). The test set perplexities of the different combinations are given in table 8. The lower the perplexity, the more predictive the dialogue model. As shown in previous dialogue modelling experiments (King, 1998; Taylor *et al.*, 1998; Chu-Carroll, 1998; Poesio & Mikheev, 1998), speaker identities are good predictors of moves in task-oriented conversations. This is the case when the different roles played by the conversational participants (*giver* and *follower* in

Utterance #	Speaker Role	Move Type	Position	Game Type
i-2	giver	instruct	start	instruct
other	follower	ready	middle	instruct
i-1	giver	instruct	middle	instruct
i	giver	acknowledge	end	instruct

Figure 2: Illustration of the predictors (circled) used in Model D for predicting the move and position of the current utterance (boxed)

the Map Task) lead to different distributions of move types. The 4-gram that reduces the perplexity the most (Model D) uses the move_position type of the other speaker's previous move (m_p_{other}) and the current and previous speaker type. This model is illustrated in figure 2.

5.1 Modifying the Move_position Utterance Type Set

One can see from table 6 that some combinations of move and position are infrequent, such as replies at the start of games and queries at the end of games. This results in sparse data problems, especially for the language and intonation models described below. Therefore, a modified set of move_position utterance types was derived by combining some of the less frequent categories[4]. This new set is referred to as MOVE_POSITION SET 2 and contains 19 categories.

A complete list of move_position set 2 types is given in table 9. The *end* and *middle* moves are combined for the following move types: *instruct*, *query-w*, *query-yn* and *ready*. This is motivated by the lack of game final utterances of these types. The start and middle categories are merged for the following move types: *reply-n, reply-y* and *acknowledge*. This is motivated by the lack of game initial utterances of these move types.

The following moves are not distinguished by their game position: *align, check, clarify, reply-w,* and *explain*. These moves have a longer, more varied syntax. Language modelling experiments described below show that it is beneficial to use one category for these utterances as this allows more data for training the models. Shorter, less varied utterance types such as acknowledgements and replies need less data for training, for example positive replies usually contain one of a small set of words "yes, yep, yeah, etc.". The utter-

Start	Middle	End
acknowledge_middle		acknowledge_end
align		
check		
clarify		
explain		
instruct_start	instruct_middle	
query-w_start	query-w_middle	
query-yn_start	query-yn_middle	
ready_start	ready_middle	
reply-n_middle		reply-n_end
reply-w		
reply-y_middle		reply-y_end

Table 9: Move frequencies with respect to game position

Model	Predictors	Perplexity
A	unigram	14
B	m_p_{i-1}	8.3
C	m_p2_{i-1}, s_i, s_{i-1}	6.9
D	m_p2_{other}, s_i, s_{i-1}	4.5

Table 10: move_position set 2 perplexity results for the different dialogue models

ance type recognition baseline is lower than the original move_position set; the most frequent move is *acknowledge_end*, which makes up 13% of the data.

5.2 Dialogue Models for move_position set 2

A number of dialogue models were developed to predict the move_position set 2 utterance types. The perplexity of the test set using these models is given in table 10. Again, the best perplexity result (4.5) is achieved by using the other person's previous utterance type (m_p2_{other}) and speaker identities (model D). This new dialogue model D was used in conjunction with the intonation model and language models in the experiments described below.

6 Game Position and Intonation Modelling

Previous studies have shown that intonation can be indicative of the position of a move in a game. For example, Nakajima & Allen (1993) show that average F0 at the start and end of an utterance varies depending on whether the utter-

ance is continuing or introducing a new topic. This suggests that moves of the same type may differ in intonation depending on their position in the game. If an utterance is game initial it may be introducing a new goal or topic and have a slightly higher utterance initial F0 contour. In order to investigate this potential correlation, we trained statistical intonation models to distinguish the combined move and position utterance types.

Wright (1998) describes three methods for modelling intonation using stochastic intonation models: hidden Markov models, classification and regression trees (CART), and neural networks. As she concludes that CART trees are slightly more effective than the other two systems, we adopted this method in our experiments here. Forty-five suprasegmental and durational features were used to construct tree structured classification rules, using the CART training algorithm (Breiman *et al.*, 1984). The tree can be examined to determine which features are the most discriminatory in move classification.

The output of the classification tree is the probability of the move (M) given the observed intonation features (I), i.e. the posterior probability $P(M|I)$. However, in order to be able to use the output of the CART model in the system described in section 3, we need to derive the likelihood $P(I|M)$ rather than the posterior probability. This can be calculated in two ways. Firstly, one can train the CART tree on equal numbers of each utterance type. A second method is to divide the posterior probability by the prior probability $P(M)$. These two methods produce similar results.

6.1 Intonation Features

The suprasegmental features are automatically extracted from the speech signal and used to train the classification tree. For each move the last three accents (if present) are automatically detected using a method described in Taylor (2000). This method identifies accents (a) and rising or falling boundary tones (rb/fb). In order to determine the type of the accents, they are automatically parameterised into four continuous *tilt parameters*: start F0, F0 amplitude, accent duration and *tilt*. Tilt is a figure between -1 and 1 and describes the relative amount of rise and fall of the pitch contour for an accent. Examples of varying

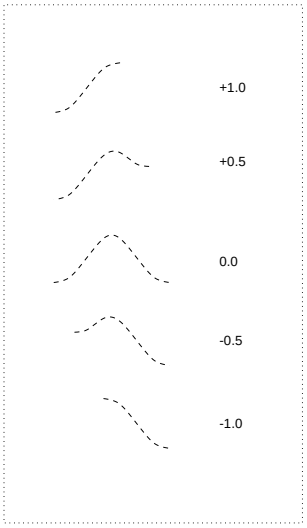

Figure 3: Values for tilt for various shaped intonation events

tilt values are given in figure 3 taken from Taylor (2000).

A set of more global features based on the study by Shriberg *et al.* (1998) is also extracted. These are prosodic features based on F0 (e.g., max F0, F0 mean and standard deviation), root mean squared (RMS) energy (e.g., energy mean and standard deviation) and duration (e.g., number of frames in utterance, number of frames of F0). These features are calculated for the whole utterance, for example, the standard deviation of the F0 represents pitch range. The least-squares regression line of the F0 contour is also calculated. This captures intonation features such as declination over the whole utterance. In addition, the above-mentioned features are calculated for the final and penultimate part of the intonation contour which is often indicative of utterance type. For example, the least square error for F0 in the final part of the contour is indicative of the type of boundary tone. Other features are calculated by comparing feature values for the last two regions and the whole utterance (e.g., ratio of mean F0 in the end and penultimate regions, difference between mean RMS energy in the end and penultimate regions). A comprehensive list of these features is given in Appendix A.

It is useful to know which features are the most discriminatory in the classification of the moves. As the tree is reasonably large with 30 leaves, interpretation is not straightforward. For simplicity, we group the features into three

Feature Type	**Usage (%)**
Duration	47
F0	41
RMS Energy	12

Table 11: Discriminatory features and type usage in move classification

general categories: duration, F0 and energy. Table 11 gives the *feature usage frequency* for these groups of features. This measure is the number of times a feature is used in the classification of data points of the training set. It reflects the position in the classification tree as the higher the feature is in the tree, the more times it will be queried. The measure is normalised to sum to 100% for the whole tree.

Different move types by their nature vary in length, so it is not surprising that duration is highly discriminatory in classifying utterance types. For example, *ready, acknowledge, reply-yes, reply-n* and *align* are distinguished from the other moves by the top node which queries a duration feature. This duration feature, regr_num_frames, is the number of frames used to compute the F0 regression line for a smoothed F0 contour over the whole utterance. This is comparable to the study reported in Shriberg *et al.* (1998), where durational features were used 55% of the time and the most queried feature was also regr_num_frames. This feature is an accurate measure of actual speech duration as it excludes pauses and silences.

The F0 features that are used frequently in the tree are F0 mean in the end region, maximum F0 and tilt value of the last accent. For example, in one part of the tree *align* moves are distinguished from *instruct* moves by having a higher F0 mean for the end region which may indicate boundary tone type.

6.2 Classification Results using the Intonation Model

A classification tree was trained on the features mentioned above to distinguish between the 19 categories in table 9. The results of these recognition experiments are given in table 12. Using the intonation model alone achieves a recognition rate of 30%, which is significantly higher than the baseline (13%). Dialogue model D has a recognition rate of 25%. Combining the intonation

	Original moves %	**move_position set 2 %**
Baseline	24	13
Intonation Model	45	30
4-gram	37	25
4-gram & Intonation	47	37

Table 12: % of utterances correctly recognised for move and move_position set 2 utterance types using Model D and intonation models

and dialogue models yields 37% correct. This is a 12% increase over the dialogue model alone.

The effectiveness of intonation models is very hard to judge. However, as the recognition results are well above the baseline, one can assume that they incorporate some of the distinguishing characteristics of the different utterance types.

7 Game Position and Language Modelling

Taylor *et al.* (1998) trained separate language models for utterances of each move type, thus capturing the lexical characteristics of each type. They show that by using these move-specific language models they can reduce the perplexity of word sequences in comparison with a general language model[5]. These language models are used to determine the likelihood of an utterance belonging to one type or another. As discussed in section 3, this is achieved by running the recogniser 12 times using each of the language models, and then choosing the move type whose associated language model produces the highest probability.

Language models are smoothed with a general model. This compensates for sparse data while still capturing the characteristics of the specific move types. For each move the perplexities of the general, move specific and smoothed models are compared and the lowest one is chosen. This result is known as the *best choice* result.

Similar language modelling experiments were run for move_position and move_position set 2. Our language models were trained using the CMU Language Modelling toolkit (Rosenfeld & Clarkson, 1997). Similar word se-

quence perplexity results were obtained for the best choice language modelling experiments. Using the original move type language models yields a perplexity of 23.8[6] whereas the move_position set 2 yields 23.9. This is promising given that the second set contains more moves and therefore there is less data to train the individual models. Using a general language model yields a higher perplexity result of 27.6.

7.1 Word Error Rate

Dialogue act recognition using the move_position set 2 labelling scheme was performed using the method described in section 3. Word error rate was therefore also calculated using move_position set 2 language models. These language models were not as useful for word recognition as the original move type set. Taylor *et al.* (1998) show that if the system could predict the correct type of utterance 100% of the time, then using the move type-specific language models was beneficial. However, this is not the case using the move_position set 2 utterance types, as the word error rate would still be above the baseline (27.7% compared to 26.1%). Using the predicted utterance type yields a word error rate of 27.6%, which again is above the baseline figure created using a general language model[7]. In other words, the reduction in word perplexity over the baseline discussed in the previous section is not always reflected in a reduction in word error rate.

However, we believe that perplexity is a better representation of what the language models are capable of as it is not affected by the idiosyncrasies of the speech recogniser. For example, if there is a high frequency of a word (such as a landmark) that the recogniser cannot recognise this will increase the word error rate.

Further experiments were run using the original move language models for word recognition but using the predicted move_position set 2 to determine the move type. This results in a similar word error rate to that reported in Taylor *et al.* (1998). This word error rate is 23.7% compared to a baseline result of 24.8%[8]. The move type recognition results are presented in the following section.

	Models Used for Move Recognition	**% Correct for Move Recognition**	**% Correct for Move Recognition collapsing m_p2**
A	Baseline	24	24
B	DM only	37	37
C	Recogniser output and LM	40	45
D	Recogniser output and LM and DM	57	64
E	IM	42	43
F	IM and DM	47	50
G	DM, IM, recogniser output and LM	**64**	**66**

Table 13: Move detection accuracy using various information sources

8 Move Recognition Results

As discussed above, the method for move recognition presented in this paper involves two stages. First, we automatically determine the likelihood of each move_position set 2 utterance type. The classification of utterances in terms of this utterance type scheme is 49%, with a baseline figure of 13%, which is the classification accuracy if *acknowledge_end* is chosen 100% of the time.

The move_position set 2 utterance types are then collapsed to obtain the likelihood of each move type. Table 13 gives the results for move classification after the move_position set 2 utterance labels have been collapsed. With the exception of experiment B (in which only the dialogue model is used), all the recognition results are increased using the new utterance types that encode the position in the game. The system as a whole increases its accuracy from 64% to 66%. Although this increase is small, it is found to be significant by a Sign test (Siegel & Castellan, 1988) ($p < 0.01, d.f. = 1060$)[9].

The confusion matrix of moves correctly recognised by the whole system is given in the matrix in table 14. The final column in this table gives the percentage of moves correctly recognised by the system that does not use

	acknowledge	align	check	clarify	explain	instruct	query-w	query-yn	ready	reply-n	reply-w	reply-y	Correct %	Original %
acknowledge	**232**	1	0	0	1	0	0	1	13	0	1	10	**90**	80
align	9	**6**	2	1	1	8	1	6	21	0	1	0	**11**	3
check	7	1	**30**	0	4	2	1	16	1	1	1	3	**45**	41
clarify	0	1	0	**5**	0	14	1	0	0	0	3	0	21	**25**
explain	8	2	6	1	**53**	15	2	7	1	3	5	2	**51**	37
instruct	1	1	3	1	9	**171**	5	3	2	0	1	3	86	**88**
query-w	3	0	1	0	3	1	**10**	2	0	1	1	2	**42**	16
query-yn	3	2	11	0	9	5	2	**50**	1	1	1	1	58	**62**
ready	32	0	0	0	1	0	0	1	**41**	1	0	2	53	**62**
reply-n	1	0	0	0	2	0	0	0	0	**25**	1	0	**86**	79
reply-w	4	0	1	0	6	6	1	0	0	0	**7**	0	**28**	26
reply-y	24	0	1	0	2	3	0	4	0	1	1	**72**	67	**70**

Table 14: Confusion matrix for move type classification: 66% move recognition accuracy

position[10]. There are several noticeable differences between the two sets of results. Firstly, using position leads to fewer *acknowledges* being misrecognised as *ready* moves, as these rarely occur in the same game position. This improvement in *acknowledge* recognition also accounts for most of the significant improvement in the experiments. Carletta *et al.* (1997) show that mistaking *acknowledges* for *ready* is also a common recognition mistake made by human labellers. Other confusions that humans make include misrecognising *query-yn* as *checks*; *ready* as *reply-ys*; and *clarifies* as *instructs*. These confusions are also observed in the above matrix; however, the confusability of these move types is lower than in the original classification matrix (see Wright (1999) for details).

Another gain that comes from taking position into account is that fewer *explain* moves are recognised as replies. This is due to the fact that *explains* mostly occur game initially, whereas replies are mostly game final. There is a 28% increase in *query-w* recognition as fewer of these move types are confused with *acknowledges*. These improvements are attributable to the dialogue model component as these move types rarely occur in the same game position. On the other hand, the dialogue model confuses more *query-yn* moves with

explains as the majority of both these move types are game initial.

There is an increase in *ready* moves that are misclassified as *acknowledges* despite the fact that they rarely occur in the same game position. On examination of the separate components, we find that this is due to the fact that the language models have a high weighting and both move types have similar wording, i.e. mostly "okay". The intonation models alone have a higher recognition accuracy for *ready* moves (64%).

Using position does not make much difference in recognising replies. This is because they have a fixed syntax that does not vary much across game position.

8.1 Declarative and Interrogative Recognition

In some cases, simpler dialogue act classification than the one attempted here is needed, such as determining whether an utterance is a question or a statement (Terry *et al.*, 1994). Experiments were conducted that examined our system's performance in making the distinction between interrogatives and declaratives. Move types considered as declaratives include *clarify, explain, instruct, reply-w*, whereas *check, query-yn, query-w* were considered as interrogatives. We also used a third category to cover the short reply type utterances, i.e. *acknowledge, align, ready* and replies. The data consists of 33% declaratives, 22% interrogatives and 45% short replies.

Figure 4 illustrates the recognition results of the three utterance types. Firstly, one can see that better results are obtained if these classifications are computed by collapsing the move_position set 2 categories rather than by collapsing the original move set; compare 79% with 84% for declaratives and 64% with 70% for interrogatives. The increase in declarative recognition is significant ($p < 0.01$). Recognition accuracy of the final category of move types was already very high with the original move type set (93%); no increase was obtained by predicting move and position simultaneously.

One can see from figure 4 that the intonation models are better than the other individual models at recognising the declarative type utterances (75%). On the other hand, our intonation models are unable to recognise interrogatives

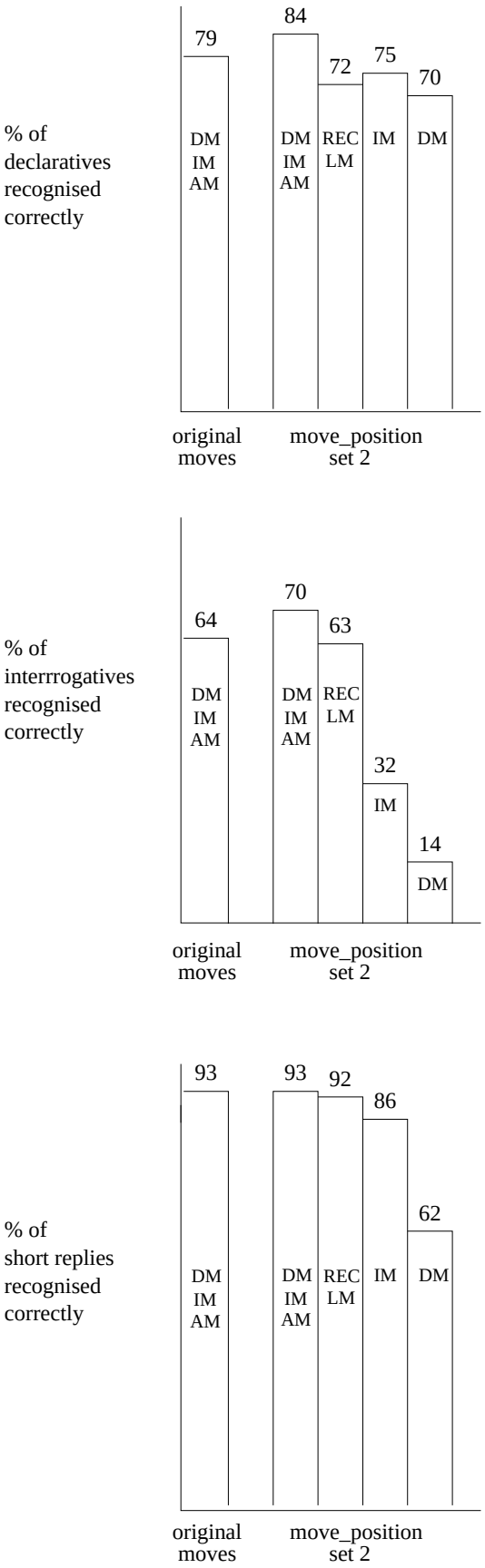

Figure 4: Percentage of interrogative, declarative and short replies correctly recognised using different knowledge sources, calculated by collapsing either the original move types (1st column) or move_position set 2 utterance types

to the same degree of accuracy (32%). One can infer from these figures that the intonation of a declarative type utterance is indicative of its function in the dialogue, but interrogative type utterances are harder to recognise using intonation alone.

Speech recognition with specific language models performs better than either intonation or dialogue models on their own for interrogative type utterance (63%). This is understandable as there is a characteristic set of words, such as "which, how, etc.", that are used in questions. Recognising declaratives, on the other hand, is more difficult as there are no keywords that indicate a declarative type utterance.

The dialogue model alone has good declarative recognition (70%) as it assigns the most common move for the follower and the giver each time. These are *instruct_inter* and *acknowledge_inter* respectively. As the model rarely predicts a question type move, the interrogative recognition is poor (14%).

The intonation models are good at recognising the third group of utterance types (86%). This is mostly due to the fact that these utterances are of similar length. As discussed above length is an important feature in the intonation model. The recognition output and language models are also very good at recognising this utterance type (92%). This is due to the similar lexical content of these utterances, i.e. mostly "okay" and either positive or negative replies.

9 Conclusion

We studied the relationship between move type prediction and game structure, in particular the position of an utterance in the game. Move type and game position were predicted simultaneously using three statistical models: dialogue, intonation and language models in conjunction with the recogniser output. Incorporating hierarchical dialogue information into the system resulted in a statistically significant improvement in move recognition. In addition to the original move types, utterances were grouped into more general categories: declaratives, interrogatives and short replies. The classification of utterances into declaratives using game position also resulted in a significant improve-

ment in recognition accuracy. An increase was also obtained for interrogative recognition.

One issue with a system such as the one discussed above is that the results are very dependent on the discourse analysis theory adopted. The discussions above have shown how difficult it is to develop models that capture both the syntactic and intonation similarities of utterances. One area of future development would be the automatic clustering of utterances by calculating some measure of distance between vectors of words or intonation features. This may result in more distinctive language and intonation models.

Another approach would be to develop context dependent categories. A study conducted by Hockey *et al.* (1997) indicates that the lexical content of a move can be predicted to a certain extent depending on the previous move. For example, there is a low probability of the word "no" if the move is preceded by an *align* move. One can hypothesise that if this is the case, then the move will be intonationally marked. Other intonationally marked moves may be non-replies preceded by queries. Training models based on move type distinguished by their context may result in sparse data problems. As with all recognition tasks, more data would result in better trained models and improved recognition results.

In conclusion, this study is an extension of previous work and has shown that using higher level game information can significantly improve the accuracy of the system in the classification of utterances into different dialogue act types. This has a number of applications including spoken language systems, meeting summarisers, data annotation and automatic speech recognition.

Notes

[1]The experiments conducted by Taylor *et al.* (1998) use a larger set of the DCIEM corpus. This set of 40 dialogues is labelled for move type but only a subset of 25 dialogues is labelled for games.

[2]Automatic move segmentation is not performed in the experiments described below.

[3]Previous experiments confirmed that using the previous two moves did not

help, as already observed in Reithinger (1995).

[4]A number of preliminary experiments were conducted that examined language model perplexities and intonation similarities to find the classification scheme that had the most potential.

[5]Higher predictability of words is reflected in a lower perplexity.

[6]This result is not comparable with that in Taylor *et al.* (1998) where a larger training set of 40 dialogues was used.

[7]The general and utterance type-specific models were trained using the smaller data set labelled for games.

[8]The general and move type-specific models were trained using the larger data set labelled for moves

[9]The Sign test examines the utterances which are classified differently by the two systems. A positive sign is given when the new system is correct and a negative sign when the original system is correct. The null hypothesis tested is that there will be more negative signs than positive ones, positing that the original system is superior to the new system.

[10]For a complete matrix see King (1998) and Wright (1999).

Appendix: Intonation and Duration Features

These features fall into the three main categories which are described in detail in the following sections:

- F0 features
- Energy features
- Duration features

F0 features

The list of features involving F0 is given in table 15. This table gives features for the utterance as a whole and over the the two end regions. Type_boundary is a binary value given for the type of boundary (1 for rising and 0 for falling). The number of accents (a), boundary (b) and joint accent boundary tones (ab) were counted (num_acc, num_bound, num_acc_bound, total_num_abs).

Feature Name	Description
max_F0	utterance max F0
utt_F0_mean	utterance mean F0
utt_F0_sd	utterance standard deviation F0
end_F0_mean	end region F0 mean
pen_F0_mean	penultimate region F0 mean
norm_end_F0_mean	end region F0 mean normalised using the utterance mean and sd
norm_pen_F0_mean	pen. region F0 mean normalised using the utterance mean and sd
abs_f0_diff	difference between mean F0 of end and penultimate region
rel_f0_diff	ratio mean F0 of end and penultimate region
norm_f0_excursion	ratio of F0 sd of end region over utterance
utt_a, utt_b	least-squares all-points regression line over utterance
end_a, end_b	least-squares all-points regression line over end region
pen_a, pen_b	least-squares all-points regression line over pen. region
type_boundary	Type of final boundary (falling, rising)
num_acc	number of accents
num_bound	number of boundaries
num_acc_bound	number of accent and boundaries
total_num_abs	total number of accents

Table 15: F0 feature list

Energy features

A general set of features is calculated for the root mean squared (RMS) energy values. These are given in table 16.

Duration Features

There are three duration features listed in table 17. Utterance duration is the number of frames of the utterance including utterance initial and final silences and voiceless segments. F0_length is taken from the start to the end of voic-

Feature Name	Description
utt_nrg_mean	mean RMS energy in utterance
utt_nrg_sd	standard deviation RMS energy in utterance
end_nrg_mean	mean RMS energy in end region
pen_nrg_mean	mean RMS energy in pen. region
norm_end_nrg_mean	mean RMS energy in end region normalised over utterance
norm_pen_nrg_mean	mean RMS energy in pen. region normalised over utterance
abs_nrg_diff	difference between mean RMS energy at end and pen. regions
norm_nrg_diff	difference between norm_end_nrg_mean and norm_pen_nrg_mean
rel_nrg_diff	ratio of end_nrg_mean and pen_nrg_mean

Table 16: Energy feature list

Feature Name	Description
utt_duration	number of frames of whole utterance
f0_length	duration of F0 contour in seconds, including voiceless frames
regr_num_frames	number of frames of F0 contour, excluding voiceless frames

Table 17: Duration feature list

ing and includes voiceless sections. Regr_num_frames is the number of frames containing voicing, used to calculate the F0 regression line for the whole utterance.

References

Bard, E. G., Sotillo, C., Anderson, A. H., & Thompson, H. 1996. The DCIEM Map Task Corpus: Spontaneous Dialogues under Sleep Deprivation and Drug Treatment. *Speech Communication*, **20**, 71–84.

Berger, A., Della Pietra, S., & Della Pietra, V. 1996. A Maximum Entropy Approach to Natural Language Processing. *Computational Linguistics*, **22**(1), 39–72.

Breiman, L., Friedman, J., & Olshen, R. 1984. *Classification and Regression Trees*. Wadsworth International, Belmont, CA.

Carletta, J., Isard, A., Isard, S., Kowtko, J., A. Newlands, A., Doherty-Sneddon, G., & Anderson, A. 1997. The reliability of a dialogue structure coding scheme. *Computational Linguistics*, **23**, 13–31.

Chu-Carroll, J. 1998. A Statistical Model for Discourse Act Recognition in Dialogue Interactions. *Pages 98–105 of: Applying Machine Learning to Discoure Processing, AAAI'98 Spring Symposium Series*.

Hockey, B. A., Rossen-Knill, D., Spejewski, B., Stone, M., & Isard, S. 1997. Can You Predict Responses to Yes/no Questions? Yes, No, and Stuff. *Pages 2267–2270 of: Proceedings of Eurospeech-97*.

Jelinek, F., & Mercer, R. L. 1980. Interpolated estimation of Markov source parameters from sparse data. *Pages 381–397 of:* **Gelesma, E. S., & Kanal, L. N.** (eds), *Pattern Recognition in Practice*. North-Holland.

King, S. 1998. *Using Information Above the Word Level for Automatic Speech Recognition*. Ph.D. thesis, University of Edinburgh.

Nagata, M., & Morimoto, T. 1994. First steps towards statistical modeling of dialogue to predict the speech act type of the next utterance. *Speech Communication*, **15**, 193–203.

Nakajima, S., & Allen, J. 1993. A Study on Prosody and Discourse Structure in Cooperative Dialogues. *Phonetica*, **50**, 197–210.

Poesio, M., & Mikheev, A. 1998. The Predictive Power of Game Structure in Dialogue Act Recognition: Experimental Results Using Maximum Entropy Estimation. *Pages 405–408 of: Proceedings of ICSLP-98*.

Power, R. 1979. The organization of purposeful dialogues. *Linguistics*, **17**, 107–152.

Rabiner, L., & Juang, B.-H. 1994. *Fundamentals of Speech Recognition*. Prentice Hall.

Reithinger, N. 1995 (March). Some Experiments in Speech Act Prediction. *In: Proceedings of the AAAI Spring Symposium on Empirical Methods in Discourse Interpretation and Generation*.

Reithinger, N., & Klesen, M. 1997. Dialogue Act classification using language models. *Pages 2235–2238 of: Proceedings of Eurospeech-97*.

Rosenfeld, R., & Clarkson, P. 1997. *CMU-Cambridge Statistical Language Modeling Toolkit v2*. `http://svr-www.eng.cam.ac.uk/~prc14/`.

Shriberg, E., Taylor, P., Bates, R., Stolcke, A., Ries, K., Jurafsky, D., Coccaro, N., Martin, R., Meteer, M., & Ess-Dykema, C. V. 1998. Can Prosody Aid the Automatic Classification of Dialog Acts in Conversational Speech? *Language and Speech*, **41**(3-4), 439–487.

Siegel, S., & Castellan, N. 1988. *Nonparametric Statistics for the Behavioral Sciences*. McGraw-Hill.

Taylor, P. A. 2000. Analysis and Synthesis of Intonation using the Tilt Model. *JASA*, **107**, 1697–1714.

Taylor, P. A., King, S., Isard, S. D., & Wright, H. 1998. Intonation and dialogue context as constraints for speech recognition. *Language and Speech*, **41**(3-4), 493–512.

Terry, M., Sparks, R., & Obenchain, P. 1994. Automated Query Identification in English Dialogue. *Pages 891–894 of: Proceedings of ICSLP 94*.

Wright, H. 1998. Automatic Utterance Type Detection Using Suprasegmental Features. *Pages 1403–1406 of: Proceedings of ICSLP'98*.

Wright, H. 1999. *Modelling Prosodic and Dialogue Information for Automatic Speech Recognition*. Ph.D. thesis, University of Edinburgh.