# USING HIGH LEVEL DIALOGUE INFORMATION FOR DIALOGUE ACT RECOGNITION USING PROSODIC FEATURES

*Helen Wright*[1], *Massimo Poesio*[2] *and Stephen Isard*[1]

[1]Centre for Speech Technology Research, University of Edinburgh,
80 South Bridge, Edinburgh EH1 1HN
http://www.cstr.ed.ac.uk

[2]Human Communication Research Centre, University of Edinburgh,
2 Buccleuch Place, Edinburgh EH8 9LW
http://www.hcrc.ed.ac.uk

email: {*helen* [1], *stepheni* [1]}@*cstr.ed.ac.uk*, poesio@cogsci.ed.ac.uk [2]

## ABSTRACT

We look at the effect of using high level discourse knowledge in dialogue act type detection. We also look at ways this knowledge can be used for improving language modelling and intonation modelling of utterance types. We find a significant improvement of predictability of dialogue models using higher level discourse knowledge.

## 1. INTRODUCTION

This paper describes a method of dialogue act recognition that takes advantage of regularities in discourse at various levels from prosodic features to goal oriented dialogue information. Pragmatic theory (Levinson [7]) suggests that conversation follows a script and that sequences of dialogue acts are not random. For example, a query followed by a response, followed by an acknowledgement is more likely than three acknowledgements in succession.

Dialog act identification is an important part of dialog systems such as Allen et al. [1], Lewin et al. [8]. It can also be used in automatic speech recognition systems to improve word error rate, Shriberg et al. [13] and Taylor et al. [15]. These systems use dialogue models trained on previous moves to predict the current move type. These models, however, are rather shallow and do not take into account regularities in sequences of dialogue acts at different points in the conversation. It is our goal to investigate the use of higher level information to predict utterance types.

The discourse analysis theory adopted here is the move-game theory first introduced by Power [11] and developed for the Maptask corpus by Carletta et al. [5]. This method divides the conversation into different games with specific goals. There are six categories of game depending on their initial move: *instruct, check, query-yn, query-w, explain and align*. The *ready* move may start a game but is not a game type. The other 5 non-initiating moves are: *acknowledge, clarify, reply-yes, reply-no and reply-w*.

In this paper we examine whether information about move position in a game and type of game can be used to predict move type sequences. For example, a move sequence such as *explain* followed by *acknowledge* is common near the end of a game as the goal of that game is achieved. Game type information is also useful for move recognition. For example, in an *instruct* game there is a higher likelihood of finding *acknowledge* moves than in a *query-yn* game where you are more likely to find *reply-yes* or *reply-no* moves.

The game position and type of a move may give us information about word sequence regularities. For instance, a *ready* move at the start of a game contains a larger vocabulary than *ready* moves in the rest of the game, as these just tend to consist of “okay”.

Finally, we examine whether game information can be used to develop better intonation models. For example, a *ready* move at the start of a game may be more emphatic than one in the middle of a game.

Motivation for this research comes from experiments described in Poesio and Mikheev [10]. Their experiments involve dialogue act detection on Glasgow Maptask corpus using Maximum Entropy Estimation (ME) [3]. Initial experiments use the previous given move to predict the current move label. These results are improved by 30% when the correct game position and game type are also used as predictors.

In our experiments, we evaluate both the results obtained by using a more complex dialogue model taking game structure into account, and the results obtained by using Maximum Entropy Estimation and N-grams to build the models of dialogue act prediction.

## 2. DATA

The experiments reported here use a subset of the DCIEM Maptask corpus [2]. This is a corpus of spontaneous goal-directed dialogue speech collected from Canadian speakers. This Maptask corpus was chosen as it is readily available, easy to analyse, has a limited vocabulary and structured speaker roles. Each conversation has two participants each with different roles called the *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.

As described above the corpus has been analysed using the game-move theory modified for Maptask dialogues. Game and move information was hand-labelled for a set of 25 dialogues which we divide into a training set of 20 dialogues (3726 utterances) and a test set of 5 dialogues (1061 utterances). None of the test set speakers are in the training set, i.e. the system is speaker independent.

## 3. SYSTEM ARCHITECTURE

As mentioned above, this system is used in an automatic speech recogniser to reduce word error rate [15]. One of 12 language models are chosen depending on the type of utterance as predicted by an automatic move detector. The reasoning behind using utterance type specific language models is that certain word sequences are more likely to occur in utterances of a certain type. For example, a question will often start with "Do you have a".

The appropriate language model is chosen by calculating the most likely move type (M) given the suprasegmental features of the utterance (I), i.e. the utterance with the highest posterior probability, $P(M|I)$. This is calculated by taking the prior probability of the move $P(M)$ and multiplying it by the output of the intonation likelihood model $P(I|M)$ described in section 3.1. This is formalised in the Bayesian formula:

$$P(M|I) \approx P(M)P(I|M)$$

A dialogue model is trained to predict the prior probability, $P(M)$, of sequences of moves. Various dialogue models were tested; the results are reported in section 4.

As described in [15] and [6], the model trained on suprasegmental features is used in conjunction with a move detector based on the output of the recogniser. A viterbi search finds the most likely path through the dialogue model, given the observations from the suprasegmental and acoustic 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 the weighted sum of the prosodic and the acoustic models.

The following experiments deal with two different scenarios. The first is called *overhearer* where the recogniser's goal is to transcribe both participants moves and words. The second *transcript* scenario uses the hand-transcribed value of a predictor at any point in the discourse. This scenario is adopted by the experiments in [10].

### 3.1. Intonation Models

Wright [16] describes 3 methods of modelling intonation using stochastic models, namely hidden Markov models, classification and regression trees (CART) and neural networks. As she concludes CART trees are slightly more effective than the other 2 systems, we adopt this method in our experiments here. 54 suprasegmental and durational features are used to construct tree structured classification rules, using the CART training algorithm [4]. The trees 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 given the features, i.e. the posterior probability $P(M|I)$. In order to compare the trees with the HMMs, the likelihood of observing a set of features given a certain move $P(I|M)$, is calculated by dividing the output of the tree by the output of the unigram, i.e. the prior probability $P(M)$. An alternative method is to train the tree on data containing equal numbers of moves. The two methods produce similar results.

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 [14]. In order to determine the type of the accents, they are automatically parameterised into 4 continuous *tilt parameters*. These are start F0, F0 amplitude, accent duration and *tilt*. *Tilt* is a figure between -1 and 1 and describes the shape of the accent.

The other prosodic features are based on F0 (e.g. max F0, F0 mean and standard deviation), 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 capture general characteristics of the utterance, for example the standard deviation of the F0 represents pitch range.

As the final part of the intonation contour is often indicative of utterance type, similar calculations are made for the last and penultimate 200ms of the utterance (e.g. mean RMS energy in the end region normalised using the mean and standard deviation of RMS energy for the whole utterance). Other features are calculated by comparing feature values for the two end 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). In addition to these features the least-squares regression line of the F0 contour is calculated for the last 200ms and for the whole utterance. This would capture intonation features such as declination over the whole utterance, and boundary type over the final part of the contour.

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 3 general categories of duration, F0 and energy. Table 1 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 1 for each tree.

Different moves 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 [13], where durational features were used 55% of the time and the most queried feature was also regr_num_frames. This feature may be a fairer measure of actual speech duration as it excludes pauses and silences.

The F0 features that come highest up in the tree are F0 mean in the end region, maximum F0 and tilt value of the last accent. This indicates that the F0 near the end of the utterance contains important linguistic information for the distinction of utterance types.

| Feature Type | Usage (%) |
|---|---|
| Duration | 0.47 |
| F0 | 0.41 |
| RMS Energy | 0.12 |

**Table 1:** Discriminatory features and type usage in move classification

| Predictor | Symbol |
|---|---|
| Move type of current move | $m_i$ |
| Identity of speaker of current move | $s_i$ |
| Identity of speaker of previous move | $s_{i-1}$ |
| Move type of previous move | $m_{i-1}$ |
| Move type of other speaker's last move | $m_{i-j}$ |
| Position in game of previous move | $p_{i-1}$ |
| Game type of previous move | $g_{i-1}$ |

**Table 2:** Notation of N-gram predictors

# 4. DIALOGUE MODEL EXPERIMENTS

In order to use context information to calculate the prior probability of a move, we trained different types of N-grams (Jelinek & Mercer 1980). We examined various types of predictor from simple unigrams that use the distribution of the moves to more complex 4-gram models. The predictors that we examined are given in table 2 where $m_i$ is the current move being predicted.

## 4.1. Dialogue Model Perplexities

In order to determine which combination of the predictors gives us the most predictive power, we look at which reduces the perplexity of the test set the most. Perplexity is a measure of how easy it is to correctly classify a move. If all the moves were equally distributed the perplexity would be 12. As some classes are more likely than others the perplexity is less than 12 as shown by using the unigram. More complex N-grams have a higher predictability and therefore reduce this perplexity further. However, using high order N-grams is problematic due to sparsity of training data, which can actually decrease the N-gram's performance. The dialogue models tested and their perplexities are given in table 3.

Model III is the dialogue model adopted in the experiments described in [15]. This uses the identity of the speaker of the current move and the previous move ($s_i$, $s_{i-1}$) and the move type of the other speaker's last move ($m_{i-j}$). Models VII and VIII have the lowest perplexity of 4.64. Model VII uses the position in the game ($p_{i-1}$), the previous move ($m_{i-1}$) and the speaker identity of the current and previous move. Model VIII is similar but uses the game type of the previous move ($g_{i-1}$) instead of the previous move. These results show that game type and position information increase predictability.

| Model | Predictors | Perplexity |
|---|---|---|
| I | unigram | 9.2 |
| II | $m_{i-1}$ | 6.2 |
| III | $m_{i-j}$, $s_i$, $s_{i-1}$ | 5.1 |
| IV | $m_{i-1}, p_{i-1}, g_{i-1}$ | 4.9 |
| V | $m_{i-j}, p_{i-1}, g_{i-1}$ | 4.7 |
| VI | $m_{i-j}, p_{i-1}, s_i, s_{i-1}$ | 4.7 |
| VII | $m_{i-1}, p_{i-1}, s_i, s_{i-1}$ | 4.64 |
| VIII | $g_{i-1}, p_{i-1}, s_i, s_{i-1}$ | 4.64 |

**Table 3:** Perplexity results for the different dialogue models

| | model III | model VII | model VIII |
|---|---|---|---|
| DM | 52 | **55.7** | 55 |
| DM, I | 54.4 | 57.6 | **58.2** |
| DM, I,REC | 64 | **69.1** | 68.9 |

**Table 4:** Percentage of moves correct using dialogue model (DM), intonation (I) and recogniser output (REC)

## 4.2. Using Dialogue Models for Move Recognition

The lower perplexity of the new models is reflected in their move recognition results. Table 4 gives these results using different levels of information for transcribed data. As we can see the move recognition accuracy increases by adding move likelihoods from the intonation models. It increases further when information from the recogniser is used. In all these cases the dialogue models that use game type and/or position information (models VII and VIII) are more accurate than model III that does not.

# 5. DIALOGUE MODELLING USING MAXIMUM ENTROPY ESTIMATION

Here we examine the effectiveness of the Maximum Entropy Estimation method as a dialogue model compared with standard N-grams.

The Maximum Entropy Principle was proposed in [3] as an alternative way of determining the posterior probability of an hypothesis H given observations O, $p(H|O)$. The principle is based on the assumption that of the many probability distributions consistent with the information acquired from the data (summarised by the empirical probability distribution $\tilde{p}$), the best one is the one which makes the fewer assumptions or is more 'uniform'. Assuming conditional entropy H(p) as a measure of the uniformity of a distribution:

$$H(p) \equiv -\sum_{i,j} \tilde{p}(x)p(y|x)logp(y|x)$$

The Maximum Entropy Principle can be formalised as: choose the probability distribution $p_*$ that maximises $H(p)$

$$p_* = \text{argmax}_p \; H(p)$$

Many implementations of the method exist, some of which in the form of off-the-shelf packages; we used the implementation developed at HCRC by Mikheev [9].

The Maximum Entropy estimator was run using the same predictors for Model V, namely last move of the other speaker, position and game type of the previous move. The values of these predictors were taken as correct. For these experiments the data was split into nine tenths training and one tenth testing data. This method got a result of 53.7% move recognition which is comparable to the N-gram method which got 55.4% correct using the same data sets.

| | moves | move+pos | position | pos+game |
|---|---|---|---|---|
| general | 27.6 | 27.6 | 27.6 | 27.6 |
| move specific | 27.16 | 32.4 | 30.1 | 34.8 |
| smoothed | 27.2 | 27.7 | 24.8 | 24.9 |
| best choice | 23.8 | **23.6** | 24.8 | 24.6 |

**Table 5:** Perplexity of test set using language modelling

| move set | baseline percentage | number of moves | perplexity unigram | perplexity bigram |
|---|---|---|---|---|
| position | 43 | 3 | 2.9 | 2.46 |
| move | 24 | 12 | 9.2 | 6.2 |
| game | 35 | 8 | 5.3 | 3 |
| move+position | 13 | 31 | 18.7 | 9.8 |
| position+game | 23 | 18 | 14.3 | 4.7 |
| move+pos+game | 12 | 117 | 38.8 | 17.1 |

**Table 6:** Baseline and perplexity results for different move sets

# 6. LANGUAGE MODELLING USING GAME INFORMATION

Taylor et al. [15] show that by using move specific language models (LM) they can reduce the perplexity of word sequences which results in a reduction in word error rate. In order to achieve this some of the move specific LM must assign a higher probability (and hence lower perplexity) to utterances of the same type than a general language model.

Similar language modelling experiments were run with different sets of moves that incorporate game information. The simplest of these sets was *position* in game which has 3 moves: *start, end* and *middle*. This set was combined with the move type to create another set: *move+position*. A combination of *position* and game type was also examined *position+game*.

Table 5 gives the perplexity of the test set using the general model and using a specific language model depending on the type of utterance. Smoothed language models are developed to compensate for lack of data for some move types. Smoothing is achieved by weighting the move specific language model with the general model. These weights are established by using a maximum likelihood method on a held out portion of the training set. In general, there is a greater weighting on the move specific language models rather than on the general model.

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. The results presented here are not directly comparable to those presented in [15] where a larger training set of 40 dialogues was used.

The move specific and smoothed models for the original move set are lower than the new alternatives. However, there is a slightly better result for the best choice method using the *move+position* set. For the original set the general model is chosen over the smoothed and move specific model more often than the *move+position* set. In general the *move+position* specific language models are better than the original move specific language models. However, there are a few that are much worse, mainly due to lack of data, therefore causing the "move-specific" perplexity be higher.

Experiments were conducted that merge some of the *move+position* moves. For example, *instruct+inter* and *instruct+end* were combined as there are few of the latter. This approach did reduce the perplexity of the move specific result to 29.7 but failed to reduce the best choice result.

Language models were trained using the CMU Language Modelling toolkit [12].

# 7. PREDICTING GAME INFORMATION

As discussed in section 6, alternate sets of moves involving game structure may improve recognition as they reduce the test set perplexity. However, in order to use these language models we must have a way of predicting the new move types automatically. If we adopt the method described in section 3, the utterance types must be intonationally similar. In this section we examine the difficulties faced when attempting game information prediction.

Table 6 gives the baseline results for the move sets described in section 6. The baseline represents the percentage of moves that would be correctly identified if we simply labelled them with the most frequent move type. Table 6 also shows the perplexity of the different sets using a unigram and a bigram.

One can see that using an N-gram to predict *position* information alone does not result in a large decrease in perplexity. In fact the unigram and the bigram tend to just assign the most frequent move *intermediate*. Although the *move+position+game* set has a high reduction in perplexity the move set is too large to possibly form realistic intonation models.

One may hypothesise that certain moves occur in certain positions more often than others. If this is the case *move+position* would be a good move set to adopt. There are, however, more move types and a lower baseline than the original test sets. This indicates that recognising *move+position* type is a harder task. This is reflected in the move recognition results. The bigram was used in overhear mode in conjunction with the intonation models to predict *move+position*. This achieved a recognition rate of 28.5%. Although this seems poor, it is a 123% increase in the baseline result. Recognition of the original 12 move set using a bigram achieves 44.7% which is an 86% increase from the baseline.

Several more complex N-grams that predict *move+position* were tested such as previous *move+position* plus current speaker identity. However, increasing the order of N-gram results in sparsity of data problems.

# 8. DISCUSSION AND FUTURE DEVELOPMENTS

We have shown that game information is useful in language modelling by creating new move categories to reduce the perplexity of the word sequence. The problem, however, is defining new categories that are easy to recognise and are intonationally similar. Future experiments include merging and splitting of categories such as *move, move+position* and *game* to try and find such a set of move types. Defining a move set that has a lower number would enable higher order N-grams as data sparsity would not be such a problem. Increasing the training set would also increase results as games are a larger unit in the discourse than moves and therefore there are fewer examples of them to train on.

As Maximum Entropy Estimation results are comparable with N-grams, at least in the transcript scenario, it would be useful to develop a technique that enables one to use the Maximum Entropy Estimation method during a viterbi search.

Future experiments also involve examining whether distinguishing embedded games provides any further useful information.

# 9. CONCLUSION

The results presented here show that information pertaining to position in game and game type can improve the predictability of dialogue models in situations where the context is given. We have shown this through reduction in test set perplexity and increase in move detection accuracy.

# ACKNOWLEDGEMENTS

Helen Wright holds an ESPRC PhD studentship 9630715.

# 10. REFERENCES

1. James F. Allen, Lenhart K. Schubert, George Ferguson, Peter Heeman, Chung Hee Hwang, Tsuneaki Kato, Marc Light, Nathaniel G. Martin, Bradford W. Miller, Massimo Poesio, and David R. Traum. The trains project: A case study in building a conversational planning agent. *Journal of Experimental and Theoretical AI*, 7:7–48, 1995.
2. Ellen G. Bard, Catherine Sotillo, Anne H. Anderson, and M. M. Taylor. The DCIEM map task corpus: Spontaneous dialogues under sleep deprivation and drug treatment. In *Proc. of the ESCA-NATO Tutorial and Workshop on Speech under Stress, Lisbon*, 1995.
3. A. Berger, S. Della Pietra, and V. Della Pietra. A maximum entropy approach to natural language processing. *Computational Linguistics*, 22(1):39–72, 1996.
4. L. Breiman, J. Friedman, and R. Olshen. *Classification and Regression Trees*. 1994.
5. Jean Carletta, A. Isard, S. Isard, J. Kowtko, A. A. Newlands, G. Doherty-Sneddon, and A. Anderson. The reliability of a dialogue structure coding scheme. *Computational Linguistics*, 23:13–31, 1997.
6. Simon King. *Using Information Above the Word Level for Automatic Speech Recognition*. PhD thesis, University of Edinburgh, 1998.
7. S. Levinson. *Pragmatics*. Cambridge University Press, 1983.
8. I. Lewin, M. Russell, D. Carter, S. Browning, K. Ponting, and S.G. Pulman. A speech-based route enquiry system built from general-purpose components. In *EUROSPEECH 93*, pages 2047–2050, 1993.
9. A. Mikheev. Feature lattices for maximum entropy modeling. In *Proc. of ACL-COLING*, pages 845–848, Montreal, CA, 1998.
10. Massimo Poesio and Andrei Mikheev. The predictive power of game structure in dialogue act recognition: Experimental results using maximum entropy estimation. In *ICSLP'98*, 1998.
11. R. Power. The organization of purposeful dialogues. *Linguistics*, 17:107–152, 1979.
12. Ronald Rosenfeld and Philip Clarkson. CMU-cambridge statistical language modeling toolkit v2. `http://svr-www.eng.cam.ac.uk/~prc14/`, 1997.
13. Elizabeth Shriberg, Paul Taylor, Rebecca Bates, Andreas Stolcke, Klaus Ries, Daniel Jurafsky, Noah Coccaro, Rachel Martin, Marie Meteer, and Carol Van Ess-Dykema. Can prosody aid the automatic classification of dialog acts in conversational speech? *Language and Speech*, 1998.
14. Paul A. Taylor. Analysis and synthesis of intonation using the tilt model. *JASA*, 2000 forthcoming.
15. Paul A. Taylor, S. King, S. D. Isard, and H. Wright. Intonation and dialogue context as constraints for speech recognition. *Language and Speech*, 41(3-4), 1998.
16. H. Wright. Automatic utterance type detection using suprasegmental features. In *ICSLP'98*, 1998.