Automatic Identification of Experts and Performance Prediction in the Multimodal Math Data Corpus through Analysis of Speech Interaction

Saturnino Luz
School of Computer Science and Statistics
Department of Computer Science
Trinity College Dublin
Dublin, Ireland
luzs@cs.tcd.ie

ABSTRACT

An analysis of multiparty interaction in the problem solving sessions of the Multimodal Math Data Corpus is presented. The analysis focuses on non-verbal cues extracted from the audio tracks. Algorithms for expert identification and performance prediction (correctness of solution) are implemented based on patterns of speech activity among session participants. Both of these categorisation algorithms employ an underlying graph-based representation of dialogues for each individual problem solving activities. The proposed Bayesian approach to expert prediction proved quite effective, reaching accuracy levels of over 92% with as few as 6 dialogues of training data. Performance prediction was not quite as effective. Although the simple graph-matching strategy employed for predicting incorrect solutions improved considerably over a Monte Carlo simulated baseline (F_1 score increased by a factor of 2.3), there is still much room for improvement in this task.

Categories and Subject Descriptors

H.3.1 [**Information Storage and Retrieval**]: Content Analysis and Indexing; I.5.2 [**Pattern Recognition**]: Classifier design and evaluation

Keywords

Collaborative problem solving, Multimodal Math Data Corpus, vocalisation graphs

1. INTRODUCTION

Beyond speech transcription and analysis of high-level content, it has become increasingly clear that paralinguistic and multimodal features hold important clues as to the nature of group interaction in a number of contexts [5, 17, 19, 4, 12]. In learning and problem solving situations involving groups, in particular, there is evidence that low-level speech features such as speech duration, energy and prosody, and writing features such as writing rate, area, aspect ratio, and pressure are predictors of social dominance and expertise [20].

This paper describes approaches to the 2013 Multimodal Learning Analytics (MMLA) challenges which are based on simple non-verbal speech interaction features (namely, structure and amount of talk, including simultaneous talk, and silence). In brief, these challenges consist in, 1) identifying the domain expert among groups of three students working cooperatively on mathematical problem solving tasks and 2) inferring, from group interaction during a problem solving task, whether the group answered the question correctly or incorrectly. We conceptualise these challenges as categorisation tasks and refer to them simply as the "expert identification task" and the "performance prediction task", respectively. The data set employed in this MMLA challenge, the Multimodal Math Data Corpus, is described in detail in [16]. A benchmark analysis is presented in [15].

A few differences between the approach reported in this paper and other uses of low-level speech features in prediction tasks, such as in speech act classification [21] and detection of decision points [6], among others, should be pointed out. First, we restrict ourselves to the above mentioned low-level speech features. No other information source, including video, writing, past performance (as in the benchmark study [15], for instance) is employed. This does not imply that we believe such features to be of lesser value for this task. Quite on the contrary. We simply wish to assess the particular contribution those speech features can make. Second, we make no use of prosodic features (speech rate, pitch, loudness) other than pause duration, though we discuss briefly how these features could be incorporated into our data representation scheme in future work (section 6). Third, we structure speech features as a vocalisation graph [5, 13] rather than consider them in isolation.

In previous research, representations based on vocalisation graphs have been successfully used, though in quite different ways, to support qualitative analyses of group behaviour [5] and clinical dialogue [7], and, in computer science, to automatically segment medical team meetings [12] and categorise patient case discussion sessions [13]. In the work reported here, we modify the underlying representation employed in

ICMI'13, December 9–13, 2013, Sydney, Australia
Copyright 2013 ACM 978-1-4503-2129-7/13/12 ...$15.00.
http://dx.doi.org/10.1145/2522848.2533788.

those works so as to abstract away speaker information. This is done so that we can deal with the performance prediction task in a completely general (i.e. session and speaker independent) way, on a nearest-neighbour framework. For the expert detection task, we derive general interaction statistics from each vocalisation graph (representing a problem-solving dialogue) and employ a strategy based on voting among naive Bayes classifiers. These approaches yielded mixed results. The speaker prediction task was quite successful within sessions. The correction prediction task was less effective. This is partly due to the imbalanced nature of the data (only 19% of answers fall into the "incorrect" category, and these are unevenly distributed across the various sessions and groups, making negative prediction harder), and partly due to the somewhat crude graph matching algorithm we employed to determine nearest neighbours. However, we believe that these problems are not insurmountable and that this graph based approach to performance prediction also shows some promise.

2. DATA PREPARATION

The original MLA corpus comprises data collected from 6 distinct, gender-matched groups of 3 students each, working on algebra and geometry problems over 12 sessions (2 separate sessions per group). Each session consisted of 16 problem solving tasks, in which the students worked cooperatively and mentored one another. The students were between 15 to 17 years old, and were grouped according to their skill levels.

The corpus made available to the participants of the ICMI MLA challenge includes, for each session: 4 audio streams (3 streams containing individual speaker audio, recorded through close-talking microphones, and 1 stream containing all audio, recorded through an omnidirectional microphone placed above the participants), 4 video streams (recorded through 3 individual cameras and 1 wide-angle camera, available in both high and low resolution versions), 3 digital pen log files (1 per student), and annotated metadata. The annotation includes, for each of task: time boundaries (start and end of problem solving, start and end of "moment of insight", time to solution), performance data (time to solution and correctness), participation data (student who initiated the answer, and the student who explained the answer, when prompted), and coding of the digital pen data.

At the session level, the metadata provided with the corpus includes the identities (participant codes) of the group leaders and dominant experts for each session. A participant's expertise is defined in terms of a cumulative score of points assigned to correctly and incorrectly solved problems, weighted according a discrete difficulty level scale ("easy", "moderate" , "hard"). The session expert is the student with the highest such expertise score. A detailed description of the tasks, the participant profiles, the data collection procedure and the data set can be found in the accompanying documentation [16].

The MLA corpus provides a rich resource for the analysis of different aspects of interaction in educational settings. However, for the purposes of the work reported in this paper, only the individual (close-talking) audio streams, the task-level timing information, and the correctness and expertise coding (to identify the target categories for classification) have been used. The processing of these data and the resulting new data sets are described next.

2.1 Generating vocalisation graph data sets

Initially, the audio files were processed in order to detect speech intervals (vocalisations) and to assign these intervals unique speaker identifiers (or the identifier "Group vocalisation" for intervals containing overlapping speech by two or more speakers). Since the individual recordings were fairly free from noise, we could employ a simple voice activity detection algorithm to generate time stamps for vocalisation intervals. For each speaker's audio file, the signal was resampled at 100ms intervals, and for each sample we marked it as a potential vocalisation (and stored it) if it exceeded an energy threshold of -36dB, otherwise it was marked as silence. Consecutive silence intervals exceeding 1s in duration indicated the end of a vocalisation, at which point the time stamps of stored vocalisation samples were smoothed and stored as an array. The arrays containing individual vocalisation profiles were then merged into a single stream of vocalisation events. From these vocalisation streams, we generated transition matrices encoding Markov chains such as the one shown in Figure 1 (top).

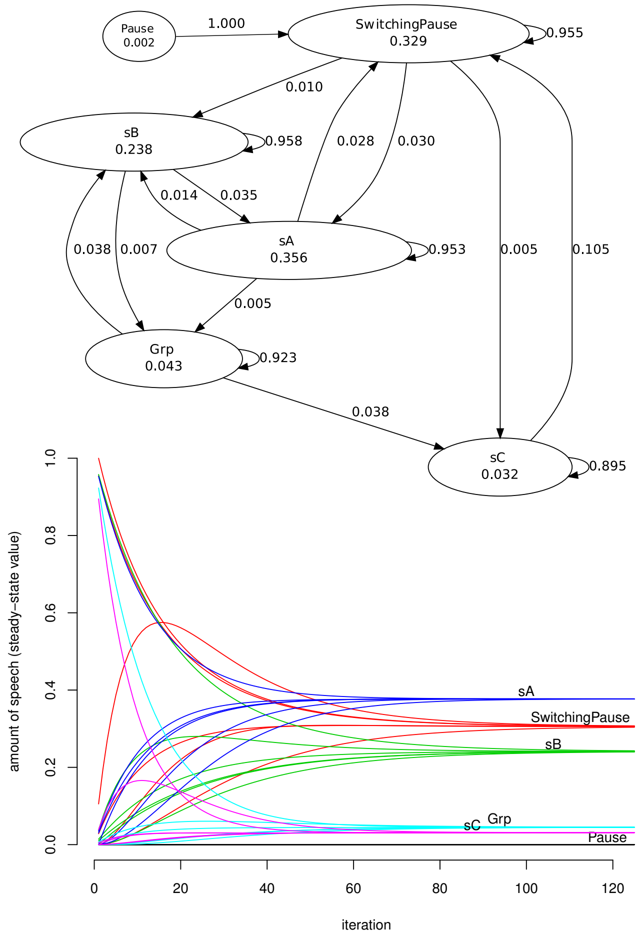

Figure 1: An individuated vocalisation graph representing the dialogue of group 1, solving problem 1A in session 1 (top) and its convergence to a steady state (bottom).

Note that very short vocalisations (< 0.9s), which for the most part corresponded to breathing noises and other non-verbal events were filtered out. This should not be taken to imply that we consider *all* such non-verbal events irrelevant. In fact, there is evidence (from other domains) that non-verbal speech sounds such as laughter and "fillers" have clear roles in structuring the dialogue [4, 2]. On the other hand, those very short bursts of audio activity would have had a detrimental effect on categorisation had they not been removed. This effect has also been observed in topic segmentation tasks [11].

A vocalisation event can be: a *vocalisation*, containing speech a participant's speech turn, a *pause*, or an interval greater than 1s in which all participants have fallen silent (we further distinguish "switching pauses", or silence interval between vocalisations by different speakers [5]), or a *group vocalisation*, where two or more speakers speak at the same time. See [12] for a more formal definition. Each entry in the vocalisation matrix represents the probability $P(V_j|V_i)$ that a vocalisation event V_i will be followed by a vocalisation event V_j, where typical transitions include, for instance, a speaker s starting to speak after a pause, a speaker s initiating a turn and being interrupted by speaker t, an interval of silence followed by two speakers speaking at the same time, and so on.

A vocalisation graph is a Markov chain containing a single aperiodic recurrent class, which means that the transition probabilities always converge to steady state probabilities by iterating the Chapman-Kolmogorov equation, as the number of iterations increases. Figure 1 (bottom) shows this convergence for the corresponding vocalisation graph (shown at the top). We define two kinds of vocalisation graphs: "individuated graphs" (like the one shown in Figure 1) and "aggregated graphs". Aggregated vocalisation graphs do not distinguish between speakers but rather pool all vocalisations into a single node.

For performance prediction (i.e. prediction of correct vs incorrect solution) we employed aggregated vocalisations graphs, since our aim has been to make this task entirely independent of sessions and groups. That is, we aim to find features that generalise across all problem solving dialogue instances. For expert identification, on the other hand, we created a feature set based on individuated graphs (omitting, of course, the specific speaker identification tags).

2.2 Expert identification data set

Expert identification posed a problem for the above-described graph representation schema. Recall that we have formulated the problem as a categorisation task. That is, given a data instance representing a speaker our goal is to label that data instance with an expert or an non-expert category tag. Therefore we cannot identify individual nodes explicitly in the representation. Neither, for the same reason, can we identify speaker to speaker transitions.

In face of this difficulty, we chose to represent each speaker s in each problem solving dialogue d as an octuple of the form:

$$s = [v_\Sigma, v_\mu, v_\sigma, p(s|f), p(f|s), p(s|g), p(g|s), H(V|s)] \quad (1)$$

where:

v_Σ is the total duration of all vocalisations produced by speaker s,

v_μ is the average duration of the vocalisations produced by speaker s,

v_σ is the standard deviation of vocalisations produced by speaker s,

$p(s|f)$ is the probability of a transition from "floor" (i.e. a pause, group switching pause or speaker switching pause) to a vocalisation produced by speaker s,

$p(f|s)$ is the probability of a transition from a vocalisation by speaker s to floor (i.e. that speaker s falls silent and no other speaker speaks for at least 1 second),

$p(s|g)$ is the probability of transitioning from a group vocalisation to a speaker s vocalisation,

$p(g|s)$ is the probability of transitioning from a speaker s vocalisation to a group vocalisation,

$H(V|s)$ is the Shannon entropy of a vocalisation event V conditioned on a speaker s vocalisation event, that is, a measure of uncertainty in the transitions (turn taking) originating from speaker s, given by

$$H(V|s) = -\sum_v P(v|s) log_2 P(v|s)$$

This representation is independent of speaker identity, group session and problem task dialogue and was generated uniformly for the entire MLA data set.

As regards our choice of features, it seems plausible that the level of expertise of a speaker (as related to their contributions to solving problems correctly) correlates with the amount of speech they produce and the regularity with which they produce their spoken contributions. Features v_Σ, v_μ and v_σ are meant to capture this putative regularity. It also seems reasonable to expect that the dominant speaker would tend to initiate turns, after moments of silence, and perhaps take most turns following a moment of simultaneous speech. Features $p(s|f)$ and $p(s|g)$ quantify such situations. Similarly, features $p(f|s)$ and $p(g|s)$ reflect the extent to which a speaker's vocalisation yields silence (perhaps indicating writing activity) and simultaneous verbal activity (perhaps indicating disagreement), respectively. Finally, the entropy feature the uniformity with which vocalisation events are distributed in succession to a speaker's vocalisation. Experts are expected to exhibit more predictable vocalisation sequence patterns (i.e. they are expected to be more consistent in their contributions). In fact, preliminary data analysis reveals a weak ($\rho = -0.18$) but reliable ($p < 0.01$) correlation between amount of speech and entropy, indicating that more verbally active speakers are also more predictable in terms of the turn taking patterns they originate.

2.3 Performance prediction data set

The performance prediction data set consists of 190 instances, corresponding to aggregated vocalisation graphs for each problem solving task dialogue. Figure 2 shows problems 1B and 2B, solved by group 1 in session 2 (G1D2), represented as vocalisation graph instances of the kind used in the performance prediction task. The graphs correspond to dialogues that resulted in a correct (problem 2B, top graph) and an incorrect (problem 1B, bottom graph) answer. A node for "switching vocalisations" was added in order to encode information about speaker change, even though the speaker identities themselves were not represented, for the reasons of generality explained above. One can see, for instance, that the incorrectly answered question produced a lot more overlapping speech (group vocalisations), less silence and less turn taking. Our working hypothesis is that these kinds of difference reflect dialogue regularities that have predictive value in this task.

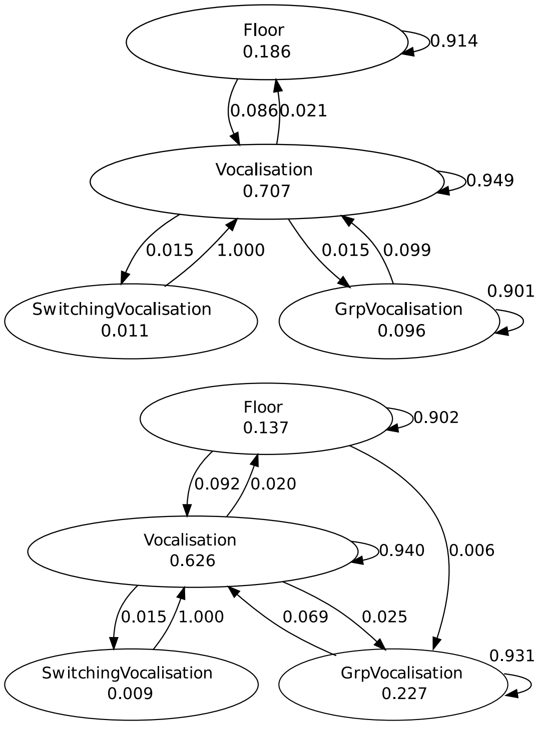

Figure 2: Aggregated vocalisation graphs showing dialogues resulting in correct (top) and incorrect (bottom) answers, for problems of similar difficulty levels.

As is the case of the representation set out in equation (1), this graph representation can be generated uniformly for the entire corpus (it can, in fact, be generated for any dialogue). However, for categorisation purposes, the matrix representation for these vocalisation graphs has been "flattened" into 40-tuple instances, each feature representing one of the $4 + 6 \times 6$ possible vocalisations and transitions, as explained in section 4.

3. EXPERT IDENTIFICATION

A continuous-variable naive Bayes approach was employed for expert prediction. For each speaker $s_a, \dots s_c$ in a problem solving task dialogue, described by features $V_1, \dots V_8$ as in equation (1), the probability that this speaker will be labelled an expert is given by equation (2), which can be rewritten as (3) under the usual conditional independence assumption.

$$
\begin{aligned}
P(t|s) &\propto P(V_1 = s_\Sigma, \dots, V_8 = H(V|s)|t) && (2)\\
&= \prod_{i=1}^{n} P(V_i = v_i|t) && (3)
\end{aligned}
$$

The estimation procedure for the conditionals in (3) models the vocalisation features through Gaussian kernels, as shown in equation (4), where μ_t and σ_t^2 are the mean and variance of the values taken by the features V_i in the data set given a positive expert label, represented here as Boolean t. Non-parametric estimation methods also exist [9] but we have not tried them in this task.

$$
\begin{aligned}
P(V_i = x|b) &= g(x; \mu_t, \sigma_t)\\
&= \frac{1}{\sigma_t \sqrt{2\pi}} e^{-\frac{(x-\mu_t)^2}{2\sigma_t^2}} && (4)
\end{aligned}
$$

These models were trained on a portion of a session's data and subsequently used to estimate probabilities for each speaker instance in the testing problem solving dialogues. Expert inference consisted in taking a vote among the estimates, so that and expert label s^* was assigned to the most likely instance:

$$
s^* = \arg \max_{s \in \{s_a \dots, s_c\}} P(t|s) \qquad (5)
$$

Similarly, one could apply a voting strategy to identify an expert for the overall session rather than individual instances by choosing the mode of the set of labels produced by evaluating all individual instances in the test set.

3.1 Expert identification results

There are different ways of going about identifying domain experts in a corpus of recorded problem solving dialogues, and at least two distinct ways of framing the problem.

As regards expert identification methods, one could base the system's inference on domain knowledge, or try to inductively learn such knowledge from relatively unstructured data. Oviatt [15], for instance, presents a rule-base method that selects as expert the participant who answers the most of the first n problems correctly (alternatively, the participant that initiates the most of the first n answers). The approach we adopted here, on the other hand, requires training data in the form of instances that describe the participants' speech interactions during a problem solving session in order to learn which instance most closely resembles the speech activity of the expert.

With respect to framing the prediction problem itself, there are, so to speak, a more general and a more specific way of looking at it. A more general perspective would define the task of the system as follows: given a single dialogue (say, a set of participant interaction instances) recorded during a problem solving activity, identify the participant who is characterised as the domain expert in the session (containing several such dialogues) during which the dialogue in question took place. This is the harder version of the problem. The specific, and somewhat easier version of the problem can be formulated as follows: given a problem solving session (as a set of dialogue representations), identify the participant characterised as the expert. The above described rule-based approach would achieve 50% accuracy [15, p. 7] in the most general formulation, since it would have to base its decision on the features of a single problem. However, in the easier formulation, the rule-based approach performs quite well, being able to correctly guess the expert after seeing only 7 problem instances.

In order to assess the performance of our approach under these different formulations of the problem we ran a series of learning iterations over the 12 sessions, training the system on different numbers of dialogues (each of which yielded 3

Table 1: Expert prediction accuracy for different numbers of training dialogues averaged over predictions for dialogues and grouped by sessions.

# training	mean accuracy	
dialogues	by dialogue(sd)	by session
1	0.54 (0.22)	0.67
2	0.58 (0.27)	0.75
3	0.55 (0.29)	0.67
4	0.60 (0.18)	0.83
5	0.61 (0.18)	0.75
6	0.62 (0.17)	0.92
7	0.70 (0.14)	1.00

training instances; one per speaker) and averaged the accuracy results over the 12 sessions. The results are shown in Table 1.

The mean accuracy by dialogue corresponds to the more general formulation. The average reported is across all sessions, and the number of training dialogues refers to the training data for each individual session. On the sixth row, for instance, the 62% accuracy score was obtained by training the system on 6 dialogues (18 speaker instances described as in equation (1)) per session, testing on the remaining 9 dialogues and averaging the results over all sessions (standard deviation of 0.17). As the training set increases in size so do the accuracy and the stability of predictions, as indicated by decreasing standard deviations.

The figures for accuracy per session correspond to the more specific formulation. As in [15] we report the accuracy scores obtained as the system "sees" progressively more dialogues of a session (though in our case the position of the problem in the session is irrelevant). For this setting we trained the models in the same way as before, but identified the expert as the one who matched the chosen expert speaker label in most dialogues of a session (or rather, most dialogues in the test set of a session). In this case, if the system is presented with, say, 6 dialogues (training set) and infers the expert for the session by choosing the majority label assigned to the remaining 10 dialogues it infers the session expert correctly for 92% of the sessions. Likewise, 7 training dialogues suffice to identify all experts correctly in all sessions. The relatively low accuracy figures in the second column (mean accuracy by dialogue) compared to the third (mean accuracy per session) reflect the fact that the former assesses classification of individual dialogues, where the system makes no use of the knowledge that a particular speaker is identified as the expert for *all* dialogues in a particular session. It is tempting therefore to speculate that in some problem solving sessions certain speakers exhibit expert behaviour even though they may not do so throughout the session.

We also ran 4-fold cross validation tests per session in order to get a better idea of expert prediction performance by dialogue (general formulation) in each session. The mean accuracy results are shown in Table 2. In most (though not all) cases, prediction performance degrades in sessions that have different speakers as assigned session leader and expert. It seems therefore that the expert leader role adds a measure of "noise" to those data folds by altering the relative amount of talk produced by non-expert leaders. However, performance in all cases is well above the 33% baseline, indicating that the method is quite robust, even in the presence of this sort of noise.

Table 2: Mean expert prediction accuracy scores for 4-fold cross-validation experiments per session.

Session	Accuracy	(sd)
G1D1	0.88	(0.34)
G1D2	0.62	(0.50)
G2D1	0.75	(0.45)
G2D2	0.69	(0.48)
G3D1	0.81	(0.40)
G3D2	0.81	(0.40)
G4D1	0.88	(0.34)
G4D2	0.69	(0.48)
G5D1	0.50	(0.52)
G5D2	0.69	(0.48)
G6D1	0.75	(0.45)
G6D2	0.62	(0.50)

4. PERFORMANCE PREDICTION

As mentioned before, the MLA corpus is quite imbalanced with respect to the questions answered correctly and incorrectly. The overall distribution is 81% for correct answers versus 19% incorrect answers. Class imbalance is known to cause difficulties to most machine learning methods [8]. To complicate matters, the balance of positive and negative instances varies greatly from session to session, reflecting the different levels of expertise of the 6 groups. Thus, the proportion of correct answers varies from 38% in G2D1 to 100% in G2D2 and G5D2.

In order to minimise this problem, we have used the entire data set for evaluation, ignoring group and session subdivisions. This also means that we attempted to model correctness and incorrectness prediction in the most general setting, in which classification does not depend on features that are specific to a group of participants. To this end, aggregated vocalisation graphs of the kind shown in Figure 2 were employed in this task.

The classification method is based on a k nearest neighbour strategy [1]. The "training" of the system consists simply in storing the training instances. Classification was performed by finding the vocalisation graph (or graphs) most similar to the test instance and assigning it the category of that training instance (or the majority label, in case there are more than one such neighbours). Instance similarity was defined in terms of the Euclidean distance

$$d(s,t) = \sqrt{\sum_{i=1}^{n} (V_i^t - V_i^s)^2}$$

between instances, taking into account the $n = 40$ features representing all transition probabilities in the graph.

4.1 Performance prediction results

The imbalanced nature of the data set must also be taken into account when assessing performance prediction results. Accuracy figures (i.e. ratio of the total number of correctly classified instances to the total number of instances tested) are typically misleading in the presence of class imbalance. For this data set, for instance, a trivial acceptor (i.e. a

Table 3: Precision, recall and F_1 results for the performance prediction task. Values denote means computed over a 10-fold cross validation.

System hypotheses			
Correct		Incorrect	
Precision:	0.850	Precision:	0.375
Recall:	0.826	Recall:	0.417
F_1:	0.838	F_1:	0.395
Baselines (Monte Carlo)			
Correct		Incorrect	
Precision:	0.805	Precision:	0.178
Recall:	0.809	Recall:	0.193
F_1:	0.802	F_1:	0.193

"classifier" that labelled all instances "correct") would have accuracy of 80% on average. We therefore provide a finer grained evaluation broken down into metrics for each class ("correct" and "incorrect") separately. These metrics include precision (ratio of the number of true positives to the total number of items categorised as the target category by the system), recall (ratio of the number of true positives to the total number of items in the target category) and F_1 (the harmonic mean of precision and recall).

Table 3 summarises the results. As expected, the "incorrect" category proved to be much harder to predict than the "correct" category. With respect to the former, the system performs quite poorly. It detects just over 41% of problems that are solved incorrectly. Furthermore, of the problems it flags as incorrect solutions, nearly two thirds are false alarms.

However, to put these results into perspective, we propose comparing them to a Monte Carlo baseline. The baseline scores shown in Table 3 were obtained by simulating 190 classification 100 times and averaging the precision, recall and F_1 results. The categorisation decisions in these simulations can be regarded as "informed guesses", since they were the result of sampling the two possible categories according to category generalities proportional to the generalities observed in the actual data set. In comparison to this baseline, the system categorisation results for the "incorrect" class represent a marked improvement, while no performance degradation is observed for the "correct" class.

It should also be remarked that the group 5 sessions are atypical in that they are extremely imbalanced. In fact, one of them (G5D2) does not contain a single error. If these sections are removed, a slight improvement in prediction performance can be observed.

In all cases, however, the models produced had fairly weak diagnostic power in general. A useful summary of overall predictive power is given by the area under the ROC (receiver operating characteristic) curve [3]. The ROC curve is a plot of the true positive rate (or recall) against the false positive rate (specificity) as the classification threshold is varied. Figure 3 shows the ROC the curves for a nearest neighbour model and a 3 nearest neighbour model, where the probabilities for the classification thresholds were obtained by weighting the neighbours proportionally to the inverse of their distance to the query instance. The diagonal represents chance classification. It can be seen from the ROC chart that the voting strategy employed by the 3-NN model improves the overall quality of the classifier, even though the improvement in terms of F-score for the negative class (incorrect answers) is negligible.

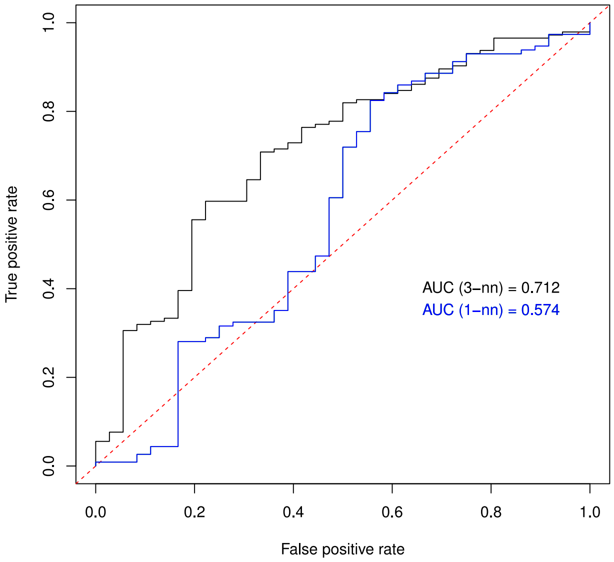

Figure 3: Area under the ROC curve for nearest neighbour (lower curve, in blue) and 3-NN (higher curve, in black) models evaluated over the 12 sessions of the MLA corpus.

5. DISCUSSION

The accuracy results obtained by the graph vocalisation method for expert identification presented in this paper are comparable to the benchmark results reported by Oviatt [15]. In fact, our results show certain advantages of our approach with respect to that benchmark. The method presented there works by keeping a count of the number of times a participant initiated an answer or correctly solved a problem. However, since the number of correct solutions forms part of the definition of expertise according to the annotation schema, and since the accuracy of solutions is directly proportional to the overall number of solutions initiated (i.e. participants who contribute many solutions are more likely to contribute correct answers than incorrect ones), classification based on these features appears to be of limited informative value.

The method described here, in contrast, only employs low-level speech features that can be easily extracted from speech recorded through close-talking microphones and require no content analysis as such. Yet, despite their simplicity, these features seem to generalise well over different groups and levels of skill. With this feature set based on vocalisation graphs, a naive Bayes classifier required very few instances to attain a reasonable level of accuracy.

The comparatively high dimensionality of the feature set used in the performance prediction task seems to have been responsible for the poorer results obtained in this task. Since the graphs had to be represented as vectors in order to allow for Euclidean distance comparison, all possible transitions had to be accommodate in the same representation. This resulted in a scheme that contained 40 features, many of which were rather sparsely filled. One of the main challenges in performance prediction is the uneven and imbalanced distribution of correct and incorrect solutions in the MLA corpus. Since detecting incorrect solution would intuitively appear to be more important from an applications perspective than simply acknowledging correct solutions, a cost sensitive classification approach may be required for this task.

The models we have tested so far essentially employ a zero-one loss function, so that classification decisions penalise all miscategorisations equally (i.e. $\lambda(T,F) = \lambda(F,T) = 1$ and $\lambda(T,T) = \lambda(F,F) = 0$, where λ is a loss function). Given that the data set is imbalanced, such a function will tend to improve accuracy for the overall session at the expense of the dialogues that resulted in incorrect answers. A cost sensitive approach would choose a threshold (defined, as illustrated in Figure 3, in terms of probabilities) so as to penalise miscategorisations of negative instances (incorrect answers) more than positive ones.

6. CONCLUSION AND FURTHER WORK

This paper reported on our initial experiments with vocalisation based ("content free") approaches to expertise detection and performance prediction in problem solving activities by small groups of students.

Overall, the results look promising. The expertise detection task was performed quite accurately based on a simple, 8-feature summary of vocalisation transitions, in problem solving dialogues, from the point of view of each speaker. The performance prediction results were poorer, but nevertheless improved upon a simulated baseline, suggesting that the graph based approach can play a role in a performance prediction method that includes other data sources. Restricted to the speech modality alone, one possible avenue for improvement would be to devise a more effective graph similarity measure to mitigate the effect of the relatively high dimensionality which seems to impair the method employed so far. We expect that this, in combination with a cost sensitive classification approach as outlined above would improve the prediction performance in this challenging task.

We also plan on incorporating richer features into the vocalisation graph framework. Speech features could be refined, for instance, by incorporating detection of laughter, fillers, hesitation markers and other non-verbal speech sounds [4]. In addition, features from the video and writing streams could also be incorporated into the data representation. It has been suggested that even fairly low-level features extracted from recorded written interaction in combination with vocalisation data can be effective in helping browsers of meeting recordings structure and visualise [10] intervals and actions of potential interest in the data [14]. Further work on multimodal learning analytics could build on this research.

7. ACKNOWLEDGEMENTS

This study was part-funded by the Science Foundation Ireland through the Centre for Next Generation Localisation (CNGL), grant number 07/CE/1142. The author also wishes to thank the anonymous reviewers of this ICMI MMLA Grand Challenge for their insightful comments and suggestions.

8. REFERENCES

[1] D. Aha and D. Kibler. Instance-based learning algorithms. *Machine Learning*, 6:37–66, 1991.

[2] F. Bonin, N. Campbell, and C. Vogel. Laughter and topic changes: Temporal distribution and information flow. In P. Baranyi, editor, *3rd IEEE Conference on Cognitive Infocommunications*, pages 53–58, 2012.

[3] A. P. Bradley. The use of the area under the ROC curve in the evaluation of machine learning algorithms. *Pattern Recognition*, 30(7):1145–1159, July 1997.

[4] N. Campbell. On the use of nonverbal speech sounds in human communication. In A. Esposito, M. Faundez-Zanuy, E. Keller, and M. Marinaro, editors, *Verbal and nonverbal communication behaviours*, Lecture Notes in Computer Science, pages 117–128. Springer, 2007.

[5] J. M. J. Dabbs and B. Ruback. Dimensions of group process: Amount and structure of vocal interaction. *Advances in Experimental Social Psychology*, 20(123–169), 1987.

[6] P.-Y. Hsueh and J. D. Moore. Automatic decision detection in meeting speech. In A. Popescu-Belis, S. Renals, and H. Bourlard, editors, *Machine Learning for Multimodal Interaction (MLMI '07)*, volume 4892 of *Lecture Notes in Computer Science*. Springer, 2007.

[7] J. Jaffe and S. Feldstein. *Rhythms of dialogue.* Academic Press, New York, 1970.

[8] N. Japkowicz and S. Stephen. The class imbalance problem: A systematic study. *Intelligent Data Analysis*, 6(5):429–449, 2002.

[9] G. H. John and P. Langley. Estimating continuous distributions in Bayesian classifiers. In Besnard, Philippe and S. Hanks, editors, *Proceedings of the 11th Conference on Uncertainty in Artificial Intelligence (UAI'95)*, pages 338–345, San Francisco, CA, USA, Aug. 1995. Morgan Kaufmann Publishers.

[10] S. Luz. Interleave factor and multimedia information visualisation. In H. Sharp, P. Chalk, J. LePeuple, and J. Rosbottom, editors, *Proceedings of Human Computer Interaction 2002*, volume 2, pages 142–146, London, 2002.

[11] S. Luz. Locating case discussion segments in recorded medical team meetings. In *Proceedings of the ACM Multimedia Workshop on Searching Spontaneous Conversational Speech (SSCS'09)*, pages 21–30, Beijing, China, Oct. 2009. ACM Press.

[12] S. Luz. The non-verbal structure of patient case discussions in multidisciplinary medical team meetings. *ACM Transactions on Information Systems*, 30(3):article 17, 2012.

[13] S. Luz and B. Kane. Classification of patient case discussions through analysis of vocalisation graphs. In *Proceedings of the 11th International Conference on Multimodal Interfaces and Machine Learning for Multimodal Interaction (ICMI-MLMI'09)*, pages 107–114, Cambridge, MA, 2009. ACM.

[14] I. McCowan, D. Gatica-Perez, S. Bengio, G. Lathoud, M. Barnard, and D. Zhang. Automatic analysis of multimodal group actions in meetings. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 27(3):305–317, March 2005.

[15] S. Oviatt. Problem-solving, domain expertise and learning: Ground-truth performance results for math data corpus. In *Second International Workshop on Multimodal Learning Analytics*, Sydney, Australia, dec 2013.

[16] S. Oviatt, A. Cohen, and N. Weibel. Multimodal learning analytics: Description of math data corpus for ICMI grand challenge workshop. Available from `http://mla.ucsd.edu/data/MMLA_Math_Data_Corpus.pdf`, 2013. Accessed August 2013.

[17] A. Pentland. Social signal processing [exploratory DSP]. *IEEE Signal Processing Magazine*, 24(4):108–111, 2007.

[18] R Development Core Team. *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna, Austria, 2011. ISBN 3-900051-07-0.

[19] S. Renals, T. Hain, and H. Bourlard. Recognition and interpretation of meetings: The AMI and AMIDA projects. In *Proc. IEEE Workshop on Automatic Speech Recognition and Understanding (ASRU '07)*, 2007.

[20] S. Scherer, N. Weibel, L.-P. Morency, and S. Oviatt. Multimodal prediction of expertise and leadership in learning groups. In *Proceedings of the 1st International Workshop on Multimodal Learning Analytics*, MLA '12, pages 1:1–1:8. ACM, 2012.

[21] E. Shriberg, A. Stolcke, D. Jurafsky, N. Coccaro, M. Meteer, R. Bates, P. Taylor, K. Ries, R. Martin, and C. van Ess-Dykema. Can prosody aid the automatic classification of dialog acts in conversational speech? *Language and Speech*, 41(3-4):443–492, 1998.

APPENDIX

A. SOURCE CODE AND DERIVED DATA SETS

The source code for the algorithms presented in this paper, and the new data and annotations produced in the course of this work are available for download from our GitLab server. The the most recent version of the source code and derived data sets that are necessary to replicate the results described above can be obtained by `git clone` 'ing the following repository:

```
http://gitlab.scss.tcd.ie/saturnino.luz/icmi-mla-challenge.git
```

The current ("frozen") version of the code and data is also available from the author's website at

```
https://www.scss.tcd.ie/~luzs/software/icmi-mla2013.tgz
```

Please note that the MMLA corpus itself is subject to a non-disclosure agreement and is therefore not distributed with the packages above. The data distributed in these packages consist solely of the derived vocalisation graphs needed to perform the categorisation experiments.

The source code has been written mostly in R [18]. It contains functions for creation and manipulation of vocalisation graphs, in addition to classification and evaluation functions. The derived data sets are encoded in standard formats used in machine learning (ARFF and CSV). The software requirements, the directory structure and a summary of the results, along with the commands used to produce them are described in `labbook/README`, in the above mentioned archive.