7th ISCA Workshop on Speech and Language Technology in Education
25-26 August 2017, Stockholm, Sweden

The University of Birmingham 2017 SLaTE CALL Shared Task Systems

Mengjie Qian, Xizi Wei, Peter Jančovič, Martin Russell

Department of Electronic, Electrical and Systems Engineering, University of Birmingham,
Birmingham B15 2TT, UK

{mxq486,xxw395,p.jancovic,m.j.russell}@bham.ac.uk

Abstract

This paper describes the system developed by the University of Birmingham for the SLaTE CALL Shared Task on grammatical and linguistic assessment of English spoken by German-speaking Swiss teenagers. Our work focused on automatic speech recognition (ASR) but we also improved the text-processing component of the system. Several approaches to training a DNN-HMM ASR system using the AMI and the German PF-STAR corpus, plus a limited amount of Shared Task data, are described. In cross-validation evaluations on the initial Shared Task data, our final ASR system achieved a word-error-rate (WER) of 9.27%, compared with 14% for the official baseline Shared Task DNN-HMM system. For text processing we expanded the baseline template-based grammar to include additional correct response patterns from the original Shared Task transcriptions. Finally, we fused the outputs of several systems at the text processing stage using linear logistic regression. Our best single and fused systems submitted to the challenge achieved 'D' scores of 4.71 and 4.766, respectively, on the final test set.

Index Terms: CALL, shared task, automatic speech recognition, text processing

1. Introduction

Since the 1980s, shared tasks have been a major factor in the development of many areas of speech and language technology, but there has not previously been such a task for Computer Assisted Language Learning (CALL). The 2017 SLaTE CALL Shared Task [1] was led by the University of Geneva with support from the University of Birmingham and Radboud University using recordings of English responses from German-speaking Swiss teenagers interacting with the CALL-SLT system [2]. A development set, ST-DEV, of 5,264 recordings, together with a true transcription, automatic speech recognition (ASR) outputs from a commercial and baseline DNN-HMM system, and a human judgment of grammatical and semantic correctness for each utterance, was released in July 2016. This set was reduced to 5,222 utterances in February 2017. This enabled participating laboratories to develop systems in time for the release of the 996 utterance test set, ST-TST, in March 2017.

This paper describes the three systems that we submitted to the 2017 SLaTE CALL Shared Task. Each consists of two components, automatic speech recognition (ASR) and text processing (TP). Our ASR system was developed using the Kaldi toolkit [3] and builds on the CALL Shared Task baseline ASR system. For ASR training, we replaced the WSJCAM0 corpus of read native English speech [4], that was used to train the baseline system, with a portion of the AMI corpus of unscripted speech [5] and the German PF-STAR corpus of German children reading English [6]. This plus 90% of ST-DEV was used for pre-training and training, followed by a final phase of training using only ST-DEV. The optimum amount of AMI training data (to balance with ST-DEV) and various parameters of the ASR system were determined empirically in cross-validation experiments on ST-DEV. For text processing we expanded the baseline grammar to include word sequence patterns from ST-DEV that were judged correct but were missing from the original grammar.

For the final evaluation on ST-TST we submitted results from three systems:

- Submission 1 consists of our best ASR system (9.27% WER average over cross-validation experiments on ST-DEV) trained on the whole of ST-DEV, plus the expanded TP. The optimal parameters of ASR for Submission 1 were estimated over 10-fold cross-validation experiments.
- Submission 2 is the result of fusing the outputs of six separate systems using linear logistic regression [7]. The systems all use our expanded TP with four variants of the ASR from Submission 1, the Kaldi baseline ASR and Nuance ASR.
- Submission 3 combines Nuance ASR with the expanded TP.

On ST-TST Submission 1 achieved a WER of 15.63% and Submission 1, 2 and 3 achieved D scores of 4.71, 4.766 and 2.533, respectively.

The rest of the paper is organised as follows. In section 2, we provide details of the spoken CALL shared task and the brief structure of our system. Sections 3 and 4 describe the ASR and text processing parts of our system, respectively. Finally, we present our conclusions in section 5.

2. Spoken CALL Shared Task

2.1. Introduction to the Shared Task

The shared task is based on data collected from CALL-SLT [8, 2], a speech-enabled online tool which has been under development at the University of Geneva since 2009. The system helps young Swiss German teenagers practise skills in English conversation. The items of data are prompt-response pairs, where the prompt is a piece of German text and the response is an utterance spoken in English and recorded as an audio file. The challenge of the task is to label pairs as "accept" or "reject", accepting responses which are grammatically and linguistically correct and rejecting those incorrect either in grammar or meaning according to the judgments of a panel of human listeners [1].

There are two versions of the task: a speech-processing version and a text-processing version [1, 9]. The aim of the two versions are the same, but they have different items provided as system input. In the speech-processing version of the CALL shared task, each item consists of an identifier, a German text

10.21437/SLaTE.2017-16

prompt and an audio file containing an English language response. For the text-processing version, there is an extra text string representing the automatic speech recognition result on the audio file, which is obtained from either the official baseline Kaldi ASR system or the Nuance ASR used in the original CALL-SLT system. This paper is mainly concerned with the speech-processing version but we also improve the text-processing version of the task.

2.2. Scoring Metric

All the items are annotated by three native English speakers according to their linguistic correctness and their meaning (these are referred to as the language and meaning "gold standard" judgments). For linguistic correctness, both vocabulary and grammar are judged as correct or incorrect. The annotators also judge whether the answer is meaningful or not in the context of the provided prompt, labelling an utterance as "sense" or "nonsense". It is worse for the system to accept a "nonsense" sentence than it is to accept one which is correct in terms of meaning. Comparing the system's judgments with the language and meaning gold standards, each response falls into one of the five categories described in Table 1.

Table 1: *Categories of Results*

English	Meaning	Judgment	Category
✓	✓	Accept	Correct Accept (CA)
✓	✓	Reject	False Reject (FR)
✗ ✓ ✗	✓ ✗ ✗	Reject	Correct Reject (CR)
✗	✓	Accept	Plain False Accept (PFA)
✓ ✗	✗ ✗	Accept	Gross False Accept (GFA)

Let CR, CA, FR, PFA and GFA denote the number of utterances in the corresponding categories as given in Table 1. The evaluation of the overall quality of the systems in the Shared Task is performed using a differential response score, D, which is defined [1] as the ratio of the reject rate on incorrect answers to the reject rate on correct utterances, i.e.,

$$D = \frac{CR/(CR + FA)}{FR/(FR + CA)} = \frac{CR(FR + CA)}{FR(CR + FA)}, \quad (1)$$

where the false acceptance (FA) is defined as $FA = PFA + k \cdot GFA$, with k being a weighting factor that causes gross false accepts to have a more prominent effect. In the current evaluation this is set to 3.

2.3. Training and Test Corpus

The training (ST-DEV) and test (ST-TST) sets of the Shared Task (ST) were released in July 2016 and March 2017, respectively. The training set contains 5,222 utterances (approx. 4.8 hours of recordings) and the test set contains 996 utterances (approx. 0.89 hours of recordings). The speakers are male and female German-speaking Swiss students ranging in age from 12 to 15 years. No specific information about speakers was released for the Shared Task, so we did not know which utterances are spoken by the same speaker or whether the speaker for a particular utterance is male or female. This has implications for ASR development because Kaldi can exploit this information if it is available. In our cross-validation experiments, the released training set ST-DEV was separated into training data and development data at the ratio of 9:1.

2.4. System Structure

The architecture of an automatic system used for the Shared Task is depicted in Figure 1. The system consists of two parts. The first is an ASR system that converts a given audio recording into a text. The second part is a text processor which takes the transcribed audio and makes a judgment of whether the utterance is accepted or rejected according to the language and meaning. A baseline ASR system built using Kaldi and a baseline text processing system were provided by the organisers of the challenge on the website [9], and we will introduce these separately in sections 3.1 and 4.1.

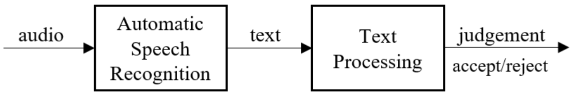

Figure 1: *Structure of the system.*

3. Automatic Speech Recognition

3.1. Official Baseline System

The provided baseline ASR system is a hybrid deep neural network – hidden Markov model (DNN-HMM) built using Kaldi [3]. The Shared Task data used to develop the baseline ASR is a super-set of ST-DEV, comprising recordings of 5,500 utterances. This corpus is referred to as ST-BASE. Thus ST-BASE includes ST-DEV plus some utterances that were not subsequently released. The baseline Kaldi ASR system is trained on about 18.93 hours of recordings from WSJCAM0 [4] and 90% of ST-BASE. The remaining 10% of ST-BASE is used for testing.

A speech signal frame is represented using 13-dimensional MFCCs with a context of 15 frames (i.e., 7 frames before and after). A neural network with 4 hidden layers and 1024 neurons for each layer was used. The output layer is a softmax layer and each node of this layer represents the posterior probability of the context-dependent HMM states. The initial training of the DNN-HMM is performed using an alignment obtained from a triphone GMM-HMM system, which was trained using an alignment obtained from a monophone GMM-HMM system. For GMM-HMM systems, a speech signal frame was represented using 13-dimensional MFCCs with delta and delta-delta coefficients appended, forming a 39-dimensional feature vector. After training the DNN model, an adaptation is applied by fine-tuning the network using only the ST data. The language model (LM) is a bigram model trained on the reference transcription of the ST data. In cross-validation evaluations, this system achieved an average WER of 14.03%.

3.2. Developed Systems

This section describes the development of ASR systems that formed part of our submission to the Shared Task challenge. All systems were developed using Kaldi. The developed DNN-HMM systems used similar configuration as the official baseline ASR system except of the following differences. We used 13-

dimensional MFCCs with context of 11 frames (i.e., ±5 frames) in most experiments – the use of a slightly smaller context than the official baseline ASR was accidental and was considered to have little effect on results. In addition, some DNN-HMM experiments (see section 3.2.2) were also performed using Mel-scaled filter-bank energies with the same 11 frames context. The neural network with 6 hidden layers and 1024 neurons for each layer was used. In all our experiments, a trigram language model trained with ST training data was used.

3.2.1. Training Data Selection

The first issue we explored was the effect of using different training data on ASR performance. The results of these experiments for monophone and triphone GMM-HMM systems and two DNN-HMM systems are shown in Table 2. In the table, "MonoPhone", "TriPhone" and "DNN" indicate systems that were pre-trained and trained on the complete training set.

In all of these experiments we also applied the fine-tuning strategy used in the baseline system, re-training the DNN model with only the ST training data after initial pre-training and training with the complete training set. "DNN.reTrain" corresponds to the same DNN-HMM system as "DNN" but after additional training using just the ST data.

Our first ASR system, Sys1 in Table 2, used only ST training data.

The WSJCAM0 corpus comprises recordings of read speech from adults who are native speakers of English. Each of these factors is inconsistent with the Shared Task data. Therefore, as an alternative, we replaced WSJCAM0 with the AMI [5] corpus in our training data. The AMI corpus consists of 100 hours of recordings of unscripted speech from adults participating in simulated meetings. Also, although the recordings are in English, English was not the first language of many of the participants. The AMI corpus was recorded using a wide range of devices, including close-talking and far-field microphone, individual and room-view video cameras. We used 77.3 hours of IHM (Individual Headset Microphone) data from the corpus in our experiments.

Although the properties of the AMI corpus are closer to those of the Shared Task, there is still a miss-match between the ages of the speakers. A model trained with AMI data will be biased towards adults' speech and will not necessarily represent the speech characteristics of young teenagers. For this reason we explored adding different amounts of AMI data to ST training data: 100% (Sys2), 50% (Sys3) and 20% (Sys4). The results are shown in Table 2.

In a further effort to incorporate more of the characteristics of the ST speakers in our training set, we also added a German English corpus, the PF-STAR corpus of recordings of read English speech spoken by German children [6], to the training set. The complete PF-STAR corpus contains more than 60 hours of speech, including read and spontaneous native language speech in British English, German and Swedish and non-native read English from German, Italian and Swedish children, aged between 4 and 15. The German part of the PF-STAR corpus (PSG) was collected from German children and includes native German recordings and non-native English recordings. The non-native English speech from PSG is used in our experiments. It contains about 3.4 hours of recordings of read speech collected from 57 German children who are aged from 10 to 15. We built a DNN system with ST, AMI and PSG training data using the same methods as those described above. The results are shown as Sys5 in Table 2.

Table 2: *%WER of development set using models trained on different training data.*

WER (%)	Sys1	Sys2	Sys3	Sys4	Sys5
MonoPhone	31.78	57.95	44.24	41.08	35.69
TriPhone	17.59	27.68	22.55	21.29	20.29
DNN	19.50	22.69	18.85	15.76	19.71
DNN.reTrain	-	14.61	13.61	13.07	14.97

Sys1: only ST training data
Sys2: ST training data plus all the AMI data
Sys3: ST training data plus 50% of the AMI data
Sys4: ST training data plus 20% of the AMI data
Sys5: ST training data plus PSG and 20% of the AMI data

From the results in Table 2, we see that adding data to the training set can improve the performance of the DNN model (although it may have disadvantages for the GMM-HMM model). Including 20% of AMI in the training set results in a greater advantage than adding all of the AMI data. The retrained DNN model for Sys2 has 14.61% WER which is better than that for Sys1 with about 25% relative improvement. Sys4 achieves a WER of 13.07%, which corresponds to a 33% improvement relative to Sys1. We expected that Sys5 would give the best performance, but it only outperforms Sys4 in the case of the GMM-HMM models and does not show an advantage for the DNN-HMM models.

3.2.2. Adaptation

This section presents application of feature normalisation and adaptation, specifically, Cepstral Mean Normalisation (CMN) and feature-space maximum likelihood linear regression (fMLLR). In Kaldi, each utterance is associated with a speaker label and consequently CMN and fMLLR are performed per-speaker. However, the speaker label information is not available in ST dataset. As such, in our Sys4 and Sys5, we used a single speaker-id for all utterances, which resulted in using a globally calculated statistics for CMN and fMLLR. In Sys6 and Sys7, we explored the application of CMN and fMLLR per-utterance basis, i.e., each utterance was considered to be from a different speaker. This was implemented in Kaldi by making the speaker-ids identical to the utterance-ids. In the case of fMLLR [10], the transformation was performed on dimensionality-reduced features. These features were obtained by first applying LDA on the 143-dimensional vector of MFCCs in context to decorrelate and reduce its dimension to 40-dimensional features and then further decorrelating using maximum likelihood linear transform (MLLT). In addition to the use of MFCCs, experiments were also performed using Mel-scaled filter-bank energies.

Experimental results for systems from 4 to 7 are presented in Table 3. It can be seen that the use of features transformed using fMLLR provides considerable performance improvements, e.g., for Sys4, from 15.76% to 13.82% and further to 10.77% after retraining. The use of per-utterance fMLLR transforms, as in Sys6 and Sys7, provided further large improvements over the use of a single global transform. The best system, using per-utterance fMLLR and the training set containing ST and 20% of AMI, achieved 8.90% WER.

3.3. Final ASR Submitted to the Challenge

In our experimental evaluations before the submission, Sys7 performed slightly better than Sys6. As such, our submissions

Table 3: *%WER of development set using models trained with mixed data (ST, AMI and PSG), the first two columns are for ST with one global speaker-id and the last two columns are for ST with different speaker-ids.*

WER (%)	Sys4	Sys5	Sys6	Sys7
DNN (fbank)	17.41	21.83	14.58	15.08
DNN (mfcc)	15.76	19.71	14.00	13.50
DNN (fmllr)	13.82	17.49	10.70	10.27
DNN.reTrain (fbank)	13.68	17.88	12.50	12.28
DNN.reTrain (mfcc)	13.07	14.97	10.95	11.78
DNN.reTrain (fmllr)	10.77	11.99	**8.90**	9.16

Sys4: ST training data plus 20% of the AMI data, one speaker-id for ST.
Sys5: ST training data plus PSG and 20% of the AMI data, one speaker-id for ST.
Sys6: ST training data plus 20% of the AMI data, different speaker-ids for ST.
Sys7: ST training data plus PSG and 20% of the AMI data, different speaker-ids for ST.

were based on Sys7. However, after the submission, we have found a minor mistake in data arrangement, which when corrected resulted in Sys6 actually performing slightly better than Sys7. Note that all results presented in this paper are after the correction was made.

The final system we used for the submission was built by following the procedure of Sys7 but using all 100% of the ST-DEV data for training the system (instead of only 90% as used in cross-validation experiments presented in the previous section). This DNN-HMM system used fMLLR-transformed features and was trained first using all the ST-DEV data plus 20% of AMI and PSG data and then further trained using only the ST-DEV data. The values for language model weight (lmwt), acoustic model weight (acwt) and insertion penalty (p) parameters were set based on best performance in our cross-validation experiments.

4. Text Processing

4.1. Official Baseline System

The official baseline text processing system, provided by the ST challenge organisers on their website [9], is based on using a reference template-based grammar. This grammar is generated based on a set of templates of responses for each prompt [11]. The baseline reference grammar includes 565 prompt-units, each prompt-unit consists of a German prompt and a set of possible responses to it. Since a German text prompt is provided for each item in the ST, we could compare the prompt with the prompt-units in the reference grammar and obtain a list of possible valid responses. If an ASR transcription of a given utterance was in the response list, then this utterance would be labelled as "accept", otherwise, it would be labelled as "reject".

4.2. Developed System

The main part of our text processing system is based on the baseline system. We expanded the reference grammar using the method described in [11], trying to make it as complete as possible. Apart from this, a pre-processing of the ASR output was included in order to deal with words due to hesitation and word repetitions, which are difficult to be handled by the grammar. An extra fusion back-end, which could take advantage of several ASR outputs, was used in our Submission 2.

4.2.1. Expanded Reference Grammar

We found that when using the baseline text processing grammar, there were a large number of false rejections on responses which seemed correct. This led us to realise that the set of responses for some prompts was not sufficient in the baseline grammar. In order to create a more complete grammar, we input the true transcriptions of speech utterances into the text processing. Theoretically, those responses which were labelled as "correct" based on the gold standard should have all been accepted. However, we have found a considerable number of rejections and these could only have been due to transcriptions not being covered by the grammar. We have also found that a few gold standard judgments were actually not correct, thus, some false rejections were actually rejected correctly.

We went through all the false rejections and added the correct transcriptions that had a correct human gold standard judgments to the grammar. We then applied text processing to the true transcriptions with the updated grammar, and went through all the false rejections and false acceptances again and updated the grammar accordingly. This procedure was repeated a few times, at each step taking care that the responses added to the grammar did not cause a large increase of false acceptances. After this, we used the actual ASR output as input to the text processing and applied the same grammar updating procedure.

We also considered adding some commonly occurring incorrect ASR outputs into the grammar in order to reduce the number of false rejections further. One such example of incorrect ASR output was that *"london eye"* was recognised as *"london a"* – such a minor error could be understood easily in real life. We considered adding these texts into the grammar, but we did not do this in our final version of grammar because of a high possibility of increasing the number of false acceptances.

4.2.2. Pre-processing

In providing a response to a given prompt, subjects may often hesitate, be uncertain or want to modify/correct their answer. These result in two main issues when assessing responses, which are difficult to remedy directly in the reference grammar. Hesitations and uncertainty may often result in insertion of words like *"um"* and *"uh"* in speech and these may appear at any place in the response. We have also noticed that responses sometimes started with the word *"hello"*. A sentence should not be rejected just because it contains such words. However, it is difficult to include these words in the reference grammar due to their possibly arbitrary location in the sentence. Thus, we removed these types of words from the ASR output before it was passed to the main text processing.

The other issue is repetition of words or modification/correction of the response. This is also not suitable to be handled by the reference grammar. We assumed that children tend to correct their response during the repetition, so the latter part of the repetition would be better. Thus, when repetition happens, we exclude the former part.

The above two steps of pre-processing provided a considerable performance improvement. However, there are still a few further issues which we have not tackled yet in our current system. One is that there are many false-start words in speech, for instance, *"i want fa five tickets"*, *"brown trou trousers"*. The reason why false-start words occur is similar to repetition, but they are harder to be excluded from the texts.

4.3. Fusion

We chose the best ASR system and the best parameters according to cross-validation experiments on ST-DEV. However, the values of the best parameters were inconsistent across different cross-validation partitions. Therefore we were not confident that the same parameters would also be optimal for ST-TST. Hence we built multiple systems, each with different ASR system parameters but the same expanded TP, and fused their outputs. We employed the weighted summation fusion approach with parameters trained on the development set to take advantage of the multiple systems.

The final output of the system is "accept" or "reject", which is a 2-class classification. For fusion we transferred the judgments into 2-class scores. Let class c_1 and c_2 represent "accept" and "reject", respectively. If the judgment for item x is "accept", then the score should be $score_{c_1}(x) = 1, score_{c_2}(x) = 0$, and if it is "reject", then the score should be $score_{c_1}(x) = 0, score_{c_2}(x) = 1$. In our experiments, we use the log score:

$$score_c(x) \leftarrow log(score_c(x) + \epsilon). \qquad (2)$$

Let there be K input systems where the ith system outputs the log score vector $score_{c,i}(X)$. Then the fused score $score_c(X)$ is given by:

$$score_c(X) = \sum_{i=1}^{K} w_{c,i} \cdot score_{c,i}(X). \qquad (3)$$

The weight, $w_{c,i}$ can be trained on the training data. After obtaining the fused score, we could assign the class for item x by:

$$class(x) = arg \max_{c} score_c(x).$$

Fusion was achieved using the linear logistic regression based fusion module in the FoCal toolkit [7].

4.4. Official Submissions

For the final evaluation on ST-TST data, we submitted results from three systems. These are summarised below together with their achieved D scores:

Submission 1 (system JJJ on the official SLaTE CALL Shared Task results table [9]) consisted of our best single ASR system and our expanded TP system. The ASR system used values for parameters (lmwt, acwt and p) that were found optimal on 10-fold cross-validation experiments. This submission achieved D score of 4.710.

Submission 2 (KKK) was the result obtained by fusing the outputs of six separate systems using linear logistic regression. The systems all used our expanded TP with four variants of the ASR from Submission 1 (with different parameter setup), the Kaldi baseline ASR and Nuance ASR. This submission achieved D score of 4.766.

Submission 3 (LLL) combined baseline Nuance ASR system with our expanded TP system. As such, this enables to evaluate the effect of our expanded TP. This submission achieved D score of 2.533.

The above results show that fusing multiple systems provided only minor performance gain over the use of the single best ASR system.

Further details of experimental results on the ST-DEV and ST-TST data are presented in Table 4. In the table, $\%Corr$ and $\%Acc$ denote percentage words correct and percentage accuracy, respectively – these were obtained using the HResults tool from HTK [12] applied to the output of each ASR system. Results show that our developed system performed considerably better on both datasets than baseline Nuance and Kaldi systems. It can also be seen that performance is considerably lower for all ASR systems on the ST-TST data compared with the ST-DEV data, indicating that there may be a mismatch between these two sets of data. The remaining lines in Table 4 present the D scores obtained by using the baseline, D_{baseTP}, and our expanded, D_{ourTP}, text processing. It can be seen that the use of our expanded grammar in the text processing can potentially have a large positive effect on the D score. The level of improvement seems to be proportional to the quality of the input passed to the text processing. For instance, on the ST-TST, the D score improved from 4.512 to 27.617 when using the true transcription, while the improvement was only by 0.175 when using Nuance ASR output (i.e., ASR whose speech recognition performance was weak).

Table 4: *Recognition performance ($\%Corr$, $\%Acc$) of ASR systems and D score for the development and test set when using true transcription and output of Nuance and Kaldi baseline recognition systems and our Submission 1 ASR system.*

	True	ASR system used		
	transc.	Nuance	Kaldi	Our-S1
ST-DEV:				
$\%Corr$	100.00	74.40	88.38	92.93
$\%Acc$	100.00	68.22	85.50	90.84
D_{baseTP}	4.231	1.950	2.278	2.779
D_{ourTP}	28.976	2.102	3.892	7.444
ST-TST:				
$\%Corr$	100.00	72.97	79.40	86.77
$\%Acc$	100.00	66.84	74.08	84.37
D_{baseTP}	4.512	2.358	1.753	2.333
D_{ourTP}	27.617	**2.533**	2.379	**4.710**

5. Conclusion

This paper has described the University of Birmingham's submissions to the 2017 SLaTE CALL Shared Task challenge. We submitted three systems, each comprising an ASR and TP component. Our initial focus was ASR and our best DNN-HMM system, developed with Kaldi using the AMI, PF-STAR (German) and Shared Task corpora, achieves WERs of 9.16% and 15.63% on ST-DEV and ST-TST, respectively. We also improved the TP component by expanding the reference grammar and pre-processing ASR output. We submitted three systems to the challenge. Submission 3 ("LLL" on the official Shared Task results table [9]), combining the baseline Nuance ASR with our expanded TP, achieves a D score of 2.533 on ST-TST. Submission 1 ("JJJ"), combining our best ASR and expanded TP, achieves a D score of 4.71. Finally, Submission 2 ("KKK") is the fusion of six separate systems, each using our expanded TP but with six different ASRs (baseline Nuance, baseline Kaldi and four variants of our best ASR system). This submission achieves the highest D score of 4.766 on ST-TST. Thus, best performance is obtained by fusing multiple complete systems. However, the performance improvement relative to the system that uses the single best ASR system is marginal.

6. References

[1] C. Baur, J. Gerlach, E. Rayner, M. Russell, and H. Strik, “A shared task for spoken CALL?” in *Proc. Language Resources and Evaluation Conf. (LREC)*, 2016.

[2] C. Baur, “The potential of interactive speech-enabled call in the swiss education system: A large-scale experiment on the basis of english CALL-SLT,” Ph.D. dissertation, Université de Genève, 2015.

[3] D. Povey, A. Ghoshal, G. Boulianne, L. Burget, O. Glembek, N. Goel, M. Hannemann, P. Motlicek, Y. Qian, and P. Schwarz, “The kaldi speech recognition toolkit,” *Proc. IEEE Workshop on Automatic Speech Recognition and Understanding (ASRU)*, 2011.

[4] T. Robinson, J. Fransen, D. Pye, J. Foote, and S. Renals, “WSJ-CAM0: A british english corpus for large vocabulary continuous speech recognition,” in *Proc. IEEE Int. Conf. on Acoustics, Speech, and Signal Proc. (ICASSP)*, 1994.

[5] J. Carletta, S. Ashby, S. Bourban, M. Flynn, M. Guillemot, T. Hain, J. Kadlec, V. Karaiskos, W. Kraaij, M. Kronenthal *et al.*, “The AMI meeting corpus: A pre-announcement,” in *Int. Workshop on Machine Learning for Multimodal Interaction*. Springer, 2005, pp. 28–39.

[6] A. Batliner, M. Blomberg, S. D’Arcy, D. Elenius, D. Giuliani, M. Gerosa, C. Hacker, M. Russell, S. Steidl, and M. Wong, “The PF_STAR children’s speech corpus,” in *Proc. European Conference on Speech Communication and Technology*, 2005.

[7] N. Brümmer, “Focal multi-class: Toolkit for evaluation, fusion and calibration of multi-class recognition scorestutorial and user manual,” *Software available at http://sites. google. com/site/nikobrummer/focalmulticlass*, 2007.

[8] E. Rayner, P. Bouillon, N. Tsourakis, J. Gerlach, Y. Nakao, and C. Baur, “A multilingual CALL game based on speech translation,” in *Proc. Language Resources and Evaluation Conf. (LREC),* Valetta, Malta, 2010.

[9] “Spoken CALL shared task official website.”

[10] S. P. Rath, D. Povey, K. Veselý, and J. Cernocký, “Improved feature processing for deep neural networks,” in *Proc. Interspeech*, 2013, pp. 109–113.

[11] E. Rayner, C. Baur, C. Chua, and N. Tsourakis, “Supervised learning of response grammars in a spoken CALL system,” in *Proc. Workshop on Speech and Language Technology in Education (SLaTE)*, 2015.

[12] S. Young, G. Evermann, M. Gales, T. Hain, D. Kershaw, X. Liu, G. Moore, J. Odell, D. Ollason, D. Povey *et al.*, “The HTK book,” *Cambridge university engineering department*, vol. 3, p. 175, 2002.