Spoken CALL Shared Task system description

Andrew Caines

Automated Language Teaching & Assessment Institute,
Department of Theoretical & Applied Linguistics, University of Cambridge, U.K.

apc38@cam.ac.uk

Abstract

We describe the systems we applied to the Spoken CALL Shared Task text-processing track in which an 'accept' or 'reject' label had to be applied to transcribed responses to a given 'prompt' stimulus. The data come from a language tutoring system for Swiss-German learners of English. In our system we attempt to capture the grammaticality and semantic dimensions of language assessment: dimensions which are explicitly labelled in the training corpus. With the training data we achieved accuracies and differential scores well above the baseline. However, our system performed less well on the test data and we discuss ways to improve performance.

Index Terms: language assessment, spoken CALL, shared task

1. Introduction

We present a system description for our submission to the Spoken CALL Shared Task, a special session at SLaTE 2017. The rationale for proposing the shared task, described in [1], was to bring together groups working on computer-assisted language learning (CALL) systems for speech, to demonstrate that the field has reached a level of maturity, and to provide a relevant and useful way of benchmarking progress in spoken CALL.

We describe our approach to the text-processing task (the one with speech transcriptions provided), explain methodological decisions, and assess system weaknesses which need to be addressed. We made two entries to the competition: one using generic features only, the other drawing on a task-specific grammar as well, named Caines_GENERIC and Caines_SPECIFIC respectively.

Results of the competition indicate that our entries ranked low, having performed reasonably well in terms of precision and recall, but being heavily penalised in the key evaluation measure for 'gross false accepts' – wrongly accepting utterances in which meaning was judged to be incorrect. We discuss ways to deal with this problem, but nevertheless propose this is a useful system design, since all features were extracted using well-established methods and open-source tools.

2. Related Work

This research relates to work on methods of linguistic feature extraction for assessment and error detection in learner language, on which there has been a multitude of work (e.g. [2, 3, 4]). Much of the literature attends to written inputs – whether that be learner essays or transcripts of speech – and depends on features which are mainly morpho-syntactic in nature: namely, word or part-of-speech *n*-grams, frequency counts, word or phrase ratios per utterance unit.

Here, we select features intended to capture the 'language' and 'meaning' dimensions of assessment in the training data. For language we turn to two measures derived from language modelling: namely, 'perplexity' scores and syntactic tree likelihoods emitted by a parser. The former may be considered a normalised measure of surprisal given an input string [5]; whilst the latter has been shown to correlate with proficiency level of the learner [6].

For meaning we adopt two measures from machine translation, designed to estimate the quality of a translated string: these are modified unigram precision, inspired by the 'BLEU score' [7], and the more refined 'Meteor' measure [8]. Finally, we calculate the cosine of prompt and response, that which has been shown to indicate the learner's adherence to the given topic [3].

3. Spoken CALL Shared Task

The shared task is described in detail in [1] and on the University of Geneva website[1]. In brief, participants had a choice of two tracks: speech processing or text processing. The former track involves speech recognition from learner recordings. In the latter track, which we selected, transcriptions are provided as the output of Nuance and Kaldi models. We opted for the Nuance data, since they appeared to be more complete without missing word '***' tokens as feature in Kaldi transcripts. In both tracks the task was the same: to automatically assign an 'accept' or 'reject' label to each input file.

The training data for the text-processing track consists of 5222 audio recordings of learners using CALL-SLT, an online tool for Swiss-German learners of English [9]. Metadata for the audio recordings lists the file identifier, 'prompt'[2], soundfile name, speech recognition output, proper transcription, and two 'correct/incorrect' labels – one for *language*, the grammar of the learner's utterance, and another for the *meaning* with reference to the prompt (Table 1).

Prompts are all of the form, *Frag:...* ('ask') or *Sag:...* ('say'), and translations were provided in the `referenceGrammar.xml` file, with allowances for justifiable variation in response. In Table 1, for example, the three prompts mean respectively: 'ask for red boots', 'ask how much it costs', 'ask for (bill | check)'. CALL-SLT users then had to do as instructed, carrying out the speech act in English in a fully constructed way.

Language and *meaning* judgements were obtained by the shared task organisers from three native speakers of English, and the set of 5222 files in the training data are the subset for which all three judges were in agreement, from a total of 6000 files judged.

Test data was in a similar form, consisting of 996 files with an identifier, prompt, soundfile name and speech recognition output for each one (Table 2).

[1] `http://regulus.unige.ch/spokencallsharedtask`

[2] In language teaching and testing, a prompt is the stimulus to which the learner must respond. It often takes the form of a question, but could alternatively be an image, audio recording, video recording, etc.

Table 1: *Spoken CALL text-processing task: training data format*

Id	Prompt	Wavfile	RecResult	Transcription	language	meaning
11336	Frag: rote Stiefel	11336.wav	i'd like red boots	i'd like red boots	correct	correct
7068	Frag: Wie viel kostet es?	7068.wav	how many is it	how many is it	incorrect	incorrect
8774	Frag: Ich möchte die Rechnung	8774.wav	i want the bill	i want the bills	incorrect	correct

Table 2: *Spoken CALL text-processing task: test data format*

Id	Prompt	Wavfile	RecResult
11336	Frag: rote Stiefel	11336.wav	i'd like red boots
7068	Frag: Wie viel kostet es?	7068.wav	how many is it
8774	Frag: Ich möchte die Rechnung	8774.wav	i want the bill

The challenge faced by participants in the shared task was to add a column to the test metadata (e.g. Table 2) indicating whether the file should be accepted or rejected, taking the given prompt and response into account. In this respect the *language* and *meaning* dimensions, distinct in training, were flattened into a single binary decision. Nevertheless the distinct dimensions played an important role in evaluation.

3.1. Evaluation

The metric for evaluation submissions to the shared task is a differential response score (D) calculated as follows:

- CORRECT REJECT: the student's answer is incorrect; the system rejects (CR).
- CORRECT ACCEPT: the student's answer is correct; the system accepts (CA).
- FALSE REJECT: the student's answer is correct; the system rejects (FR).
- PLAIN FALSE ACCEPT: the student's answer is correct in meaning but grammatically incorrect. The system accepts (PFA).
- GROSS FALSE ACCEPT: the student's answer is incorrect in meaning; the system accepts (GFA).

Overall false accepts (FA) are then calculated as the sum of PFA and GFA, in which gross false accepts are heavily penalised by a weighting factor k where $k = 3$. The intuition here is that a CALL system accepting an utterance which does not make sense is a graver error than a CALL system accepting an ungrammatical utterance.

Finally, the D-score is the ratio of the reject rate on incorrect answers to the reject rate on correct utterances:

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

As shown in 1, the higher the D-score the better, as a high proportion of true rejections represents the numerator, which would ideally be divided by a small proportion of false rejections – the denominator.

4. System Description

In our approach to the shared task we had two overarching aims in mind: to do as well as possible in the task, and to build a system which would generalise to other spoken CALL scenarios beyond this task. While we prioritised the latter, since we're involved in a long-term spoken CALL project as part of the ALTA Institute[3], we of course still wanted to successfully address the given task. Thus we made two submissions both fundamentally the same but with one key difference: use of the `referenceGrammar.xml` resource, which lists permissible responses to each prompt.

We refer to our general system as Caines_GENERIC and the system referring to the grammar specific to this task as Caines_SPECIFIC. We expect the task-specific system to perform better than the general one, but the general system remains of interest to us in the big picture.

We begin with the fundamentals shared by both Caines_GENERIC and Caines_SPECIFIC:

- We **preprocessed** the training metadata (text-processing track, Nuance version) so that for each input file we had a translation of the prompt, retrieved from `referenceGrammar.xml` and an overall *judgement* label based on the *language* and *meaning* 'correct/incorrect' labels[4], in this way folding the two assessment criteria into one as required by the task.
- For **feature extraction** we tried to encapsulate the intuitions underlying the *language* and *meaning* ratings – that a good answer should be grammatical and occupying a similar semantic space to the prompt. To capture the *language* dimension we did the following:
    1. Obtained a 'perplexity' score for each input transcription: the inverse probability of the response, normalised by length, according to a language model trained on the 10-million word spoken section of the British National Corpus [10]. Perplexity scores were obtained using KenLM [11], and included out-of-vocabulary words.
    2. Obtained the likelihood score assigned by the RASP System [12] to its top-ranked parse analysis of the input transcription, normalised by the number of nodes in the tree (since parse likelihoods are

[3] `http://alta.cambridgeenglish.org`

[4] If either *language* or *meaning* is labelled 'incorrect' then the overall label is 'reject'; else it is 'accept'.

known to decrease with increasing complexity of tree structure).

   3. Obtained a measure of response length by counting the number of tokens in the input transcription. It has been shown in other work that utterance length correlates with assessment scores [13].

- We took several measures to try and capture *meaning*:
   1. Semantic similarity between prompt and response as the cosine of their word vectors over N, where N is the vocabulary of the training corpus (or the test corpus, in the test phase) and word frequencies are expressed as 'tf-idf': the product of term frequency (count in given document) and inverse document frequency ($log(\frac{n}{df})$ where n is the number of documents in the corpus, and *df* is the number of documents in which the given term occurs).
   2. Modified unigram precision of the response compared to the prompt, in a move inspired by the BLEU score used in evaluation of machine translation [7]. This is the proportion of unigrams in the response which are also found in the prompt, 'clipped' at the maximum number of times each unigram occurs in the prompt so that pathological responses (e.g. 'Boots boots boots') do not score unduly highly.
   3. Another metric borrowed from machine translation is 'Meteor' [8], which aligns translation hypotheses to reference translations and calculates a sentence-level similarity score based on exact matches, stemmed matches, synonyms from WordNet[5], and paraphrases. We treat the response as the hypothesis and the prompt as the reference in the input parallel corpus for the Meteor system.
- These features were extracted for both Caines_GENERIC and Caines_SPECIFIC. Additionally for Caines_SPECIFIC, we checked if the learner's response matched any of the possible answers given in `referenceGrammar.xml` for the binary feature *inGrammar*. We did not extend the file in any way, and heed the organisers' warning that it is incomplete, but nevertheless recognise that if the response is in the grammar it's a strong indicator of acceptability.

4.1. Implementation

We employed the aforementioned features in a binomial logistic regression model and a support vector machine. Despite many feature combinations and attempts to tune the SVM, we found logistic regression to be more accurate and therefore devote the remainder of this section to the regression model.

We use ten-fold cross-validation to segment the training corpus of 5222 items into tenths (nine segments of 522 items, one of 524), evaluating a regression model on each fold, having trained the model on the other nine-tenths of the corpus. Using R [14] we fitted a generalised linear model (family: binomial, link: logistic) to the training data with *judgement* as the dependent variable and *perplexity, parse likelihood, length, cosine, unigram precision* and *meteor* as the independent variables.

We repeatedly found through inspection of analysis-of-deviance tables that perplexity and unigram precision do not greatly improve the model and have high p-values (test: χ-squared). For the sake of model parsimony we dropped these features.

4.2. Performance on training data

Therefore with just *parse likelihood, length, cosine* and *meteor* we trained a new regression model, Caines_GENERIC, which over ten-folds returned a mean accuracy of 77.4%, where accuracy is the proportion of true labels out of all items. However, as explained in §3.1, not all false labels carry the same weight: 'gross false accepts', a false accept when the meaning is incorrect, are penalised more heavily ($*3$) than other false labels.

Table 3: *Spoken CALL text-processing task: logistic regression performance on training data (iRej: rejections on incorrect responses, $\frac{CR}{CR+PFA+(GFA*3)}$; cRej: rejections on correct responses, $\frac{FR}{FR+CA}$)*

	Baseline	Caines_GENERIC	Caines_SPECIFIC
CR	885	409	**914**
PFA	358	663	**293**
GFA	**99**	270	135
FR	1235	**250**	635
CA	2645	**3630**	3245
iRej	0.575	0.217	0.567
cRej	0.318	0.064	0.164
D	1.81	3.37	3.46

The workings for the D-score for this first system, Caines_GENERIC, are given in Table 3. We also report the performance of a baseline system provided by the organisers – one which seeks out the response in `referenceGrammar.xml`, performing at 67.5% accuracy on the training data.

We note that our system Caines_GENERIC outscores the baseline in large part thanks to a large decrease in false rejects, despite a large increase in false accepts including three times the number of GFAs. Nevertheless the results are encouraging as the method of feature extraction is transferable to other learner corpora where the prompt is known.

With our second system Caines_SPECIFIC, we restore the baseline check for a matching response in `referenceGrammar.xml`. As a result, overall accuracy increases to 79.6% and the D-score for the training corpus is reported in Table 3. We see that Caines_SPECIFIC only marginally outperforms Caines_GENERIC in terms of D-score, an improvement which comes in large part from fewer gross and plain false accepts and more correct rejects (though more false rejects). The best-performing system for each evaluation facet is shown in bold in Table 3.

Finally, we trained our two models Caines_GENERIC and Caines_SPECIFIC on the whole training corpus, using *parse likelihood, length, cosine* and *meteor*, plus *inGrammar* for the task-specific model. Coefficients and p-values are given in Tables 4 and 5.

4.3. Performance on test data

It transpires that our systems do not perform as well on the test set as they did on the training set. Table 6 shows precision, recall, F-measure (harmonic mean of p and r) and differential response score (D) for our two submissions, Caines_GENERIC

[5] `http://wordnet.princeton.edu`

Table 4: *Spoken CALL text-processing task: logistic regression feature analysis for Caines_*GENERIC*, training data*

	B	e	p
(intercept)	0.397	0.157	0.012
cosine	2.276	0.157	<0.001
length	-0.284	0.021	<0.001
p.likelihood	0.280	0.067	<0.001
meteor	2.764	0.204	<0.001

Table 5: *Spoken CALL text-processing task: logistic regression feature analysis for Caines_*SPECIFIC*, training data*

	B	e	p
(intercept)	-0.596	0.181	<0.001
cosine	1.32	0.188	<0.001
length	-0.075	0.023	0.001
p.likelihood	0.419	0.081	<0.001
meteor	1.003	0.241	<0.001
inGrammar	3.059	0.105	<0.001

and Caines_SPECIFIC. It is apparent that our system does quite well at confirming that correct responses are correct, but at the same time suffers from a generosity to incorrect responses resulting in low iRej ratios.

Table 6: *Spoken CALL text-processing task: logistic regression performance on test data (iRej: rejections on incorrect responses,* $\frac{CR}{CR+PFA+(GFA*3)}$*; cRej: rejections on correct responses,* $\frac{FR}{FR+CA}$*)*

	Baseline	Caines_GENERIC	Caines_SPECIFIC
P	0.822	0.737	0.622
R	0.723	0.820	0.897
F	0.770	0.776	0.735
CR	**210**	153	68
PFA	**49**	86	123
GFA	**21**	41	89
FR	198	129	**74**
CA	518	587	**642**
iRej	0.652	0.423	0.148
cRej	0.277	0.180	0.103
D	2.358	2.346	1.437

Compared to other entrants, our information theoretic scores (p, r, F) are good. We suffer in the D-metric from a relatively low iRej ratio (Table 6[6]). We also note that of the 14 text track entries, those who selected the Nuance transcripts (as we did) make up the bottom four; the remainder having selected the Kaldi transcripts. If we instead use the Kaldi test data, the D-score for Caines_GENERIC is 2.43 and Caines_SPECIFIC becomes 1.69 (oddly, at 1.694, the baseline Kaldi score is lower than the Nuance equivalent). Indeed with gold standard transcripts, the baseline D-score rises to 4.51, Caines_GENERIC to 2.8 and Caines_SPECIFIC to 6.39, so clearly there's a detrimental effect of ASR errors on system judgements since they all to some extent seek to match the response string with the prompt (plus the grammar in the case of the baseline and Caines_SPECIFIC).

Table 7: *Spoken CALL text-processing task: logistic regression feature analysis for Caines_*GENERIC*, test data*

	B	e	p
(intercept)	0.745	0.310	0.016
cosine	2.194	0.321	<0.001
length	-0.256	0.046	<0.001
p.likelihood	-0.210	0.148	0.157
meteor	0.618	0.422	0.143

Table 8: *Spoken CALL text-processing task: logistic regression feature analysis for Caines_*SPECIFIC*, test data*

	B	e	p
(intercept)	0.285	0.363	0.432
cosine	2.15	0.324	<0.001
length	-0.234	0.048	<0.001
p.likelihood	-0.222	0.149	0.135
meteor	0.499	0.428	0.243
inGrammar	0.487	0.195	0.012

Compared to model performance on the training data, it is apparent that the parse likelihood and Meteor features did not generalise as well as cosine and utterance length, when we train a logistic regression model on the test data only (Tables 7, 8; *cf.* Tables 4, 5). Indeed in both models parse likelihood has a negative coefficient for acceptability of response – in contrast to the training data where an increase in parse likelihood correlated with increasing acceptability of response. This may be a result of ASR transcript errors (reference transcriptions were used in training) and indicates how easily such errors cascade through a system.

5. Discussion

Since we are penalised three times more heavily for gross false accepts than any other error, there is a clear incentive to reduce their frequency as much as possible. To that end, we inspected the GFAs produced by our task-specific system Caines_SPECIFIC. Table 9 shows a sample of the 135 GFAs in the training corpus.

We found that our measures of semantic similarity and grammaticality are inadequate in several ways. First, they do not distinguish incorrect selection of *wh* question words, as in items 5841 and 5874. A dictionary would solve this problem, but this is a more task-specific solution than we'd like to adopt: not all spoken CALL systems require the learner to repeat the stimulus so closely.

Secondly, semantic errors of the kind seen in 5848 and 5984, where the learner says 'three' instead of 'two' could be addressed with richer meaning representations – obtaining a semantic parse rather than distributional or n-gram matching methods. The shallow semantic representation currently obtained also accounts for 5960 and 6147, in which grammatical words are combined in nonsensical ways. A richer semantic parse of both prompt and response would help prevent GFAs

[6] The baseline here is the supplied 'check in grammar' Python script used on the training data.

Table 9: *Spoken CALL text-processing task: gross false accepts in the training data by Caines*_SPECIFIC

Id	Prompt	Prompt_EN	Response
5841	Frag: Wo ist der Coiffeur?	Ask: where is the hairdresser?	who is the hairdresser
5848	Frag: Zimmer für 2 Nächte	Ask for: a room for two nights	room for three nights
5874	Frag: Wo ist die Hotelbar?	Ask: where is the hotel bar?	who is hotel bar
5958	Frag: Grösse 40	Ask for: a size forty	r t
5960	Frag: Stiefel	Ask for: some boots	i will boots
5984	Frag: Zimmer für 6 Nächte	Ask for: a room for six nights	i want a room for three nights
5991	Frag: Einzelzimmer	Ask for: a single room	no
6002	Frag: Gibt es ein Fitnessstudio?	Ask: is there (a fitness centre | a gym)?	there a fitness centre
6147	Sag: Ich möchte mit Dollars bezahlen	Say: i would like to pay by dollars	can i buy be dollars
6255	Sag: Ich gehe in die Ferien	Say: I am on holiday	holidays

such as these, perhaps using methods of semantic representation such as those described in [15, 16].

Finally, it is apparent from 5958, 5991, 6002 and 6255 that ungrammatical responses are not being detected adequately. A character-level language model would prevent 'r t' being labelled 'accept' (5958), and a hard-coded minimum length requirement may be worth investigating if it helps prevent GFAs such as 5991 and 6255. As for 6002, its ungrammaticality would be detected by richer syntactic features – these could be collected at the same time as the RASP likelihoods.

These new insights came after the submission deadline, but we would incorporate them in future work, along with exploration of different machine learning algorithms for the sake of improved performance and increased flexibility over logistic regression. In any case, we found the shared task a useful exercise for scenarios in which the exam or test prompt is known – as it is in our general research programme.

6. Conclusion and Future Work

In future work we may wish to consider the acoustic signal as well as text features. For example we can draw on previous experiments extracting prosodic features such as speech rate, loudness and pitch values [17]. There has also been phonological work on vowel space showing that higher proficiency learners have a more distinctive set of vowels [18]. And we can seek to represent meaning in a more formal way – for instance, using semantic graphs to verify whether the candidate has answered the question [16, 15], or coherence measures to test how well the response holds together [19].

Above all, we have endeavoured to produce an assessment system which is general purpose and uses open-source tools for two main reasons. Firstly on the grounds that it would be time-consuming and costly in less constrained domains to produce an exhaustive grammar of all possible responses as seen in `referenceGrammar.xml`. Language is known to be infinitely creative, and learners have access to options of expression which cannot be fully itemised. Secondly so that the system is reproducible as a potentially-useful baseline in other work.

7. Acknowledgements

This paper reports on research supported by Cambridge English, University of Cambridge. I'm grateful to colleagues at ALTA-DTAL and the organisers of the Shared Task for their efforts. I also thank the three reviewers for their helpful feedback.

8. References

[1] C. Baur, J. Gerlach, M. Rayner, M. Russell, and H. Strik, "A shared task for spoken CALL?" in *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC 2016)*, 2016.

[2] S. Bhat and S.-Y. Yoon, "Automatic assessment of syntactic complexity for spontaneous speech scoring," *Speech Communication*, vol. 67, pp. 42–57, 2015.

[3] K. Evanini, S. Xie, and K. Zechner, "Prompt-based content scoring for automated spoken language assessment," in *Proceedings of the Eighth Workshop on Innovative Use of NLP for Building Educational Applications*, 2013.

[4] I. Pilán, E. Volodina, and T. Zesch, "Predicting proficiency levels in learner writings by transferring a linguistic complexity model from expert-written coursebooks," in *Proceedings of COLING*, 2016.

[5] P. Brown, V. D. Pietra, R. Mercer, S. D. Pietra, and J. Lai, "An estimate on the upper bound for the entropy of English," *Computational Linguistics*, vol. 18, pp. 31–40, 1992.

[6] A. Caines and P. Buttery, "The effect of disfluencies and learner errors on the parsing of spoken learner language," in *Proceedings of the First Joint Workshop on Statistical Parsing of Morphologically Rich Languages and Syntactic Analysis of Non-Canonical Languages*, 2014.

[7] K. Papineni, S. Roukos, T. Ward, and W.-J. Zhu, "Bleu: a method for automatic evaluation of machine translation," in *Proceedings of 40th Annual Meeting of the Association for Computational Linguistics*, 2002.

[8] M. Denkowski and A. Lavie, "Meteor universal: Language specific translation evaluation for any target language," in *Proceedings of the EACL 2014 Workshop on Statistical Machine Translation*, 2014.

[9] M. Rayner, P. Bouillon, N. Tsourakis, J. Gerlach, M. Georgescul, Y. Nakao, and C. Baur, "A multilingual CALL game based on speech translation," in *Proceedings of the Seventh International Conference on Language Resources and Evaluation (LREC 2010)*, 2010.

[10] "The British National Corpus, version 2 (BNC World)," 2001.

[11] K. Heafield, "KenLM: faster and smaller language model queries," in *Proceedings of the EMNLP 2011 Sixth Workshop on Statistical Machine Translation*, 2011.

[12] T. Briscoe, J. Carroll, and R. Watson, "The second release of the RASP System," in *Proceedings of the COLING/ACL 2006 Interactive Presentations Session*. Association for Computational Linguistics, 2006.

[13] J. Kormos and M. Dénes, "Exploring measures and perceptions of fluency in the speech of second language learners," *System*, vol. 32, pp. 145–164, 2004.

[14] R. Core Team, "R: a language and environment for statistical computing," 2017.

[15] R. Ferreira, R. D. Lins, S. Simske, F. Freitas, and M. Riss, “Assessing sentence similarity through lexical, syntactic and semantic analysis,” *Computer Speech and Language*, vol. 39, pp. 1–28, 2016.

[16] S. Reddy, O. Täckström, M. Collins, T. Kwiatkowski, D. Das, M. Steedman, and M. Lapata, “Transforming dependency structures to logical forms for semantic parsing,” *Transactions of the Association for Computational Linguistics*, vol. 4, pp. 127–140, 2016.

[17] R. C. van Dalen, K. M. Knill, and M. J. F. Gales, “Automatically grading learners’ English using a Gaussian process,” in *Proceedings of the Sixth ISCA Workshop on Speech and Language Technology in Education (SLaTE)*, 2015.

[18] C. Graham, P. Buttery, and F. Nolan, “Vowels characteristics in the assessment of L2 English pronunciation,” in *Proceedings of INTERSPEECH*, 2016.

[19] K. Zupanc and Z. Bosnić, “Automated essay evaluation with semantic analysis,” *Knowledge-Based Systems*, vol. 120, pp. 118–132, 2017.