SLaTE 2019: 8th ISCA Workshop on Speech and Language Technology in Education
20-21 September 2019, Graz, Austria

The FBK system for the 2019 Spoken CALL Shared Task

Roberto Gretter, Marco Matassoni, Daniele Falavigna

Fondazione Bruno Kessler, Trento (Italy)
{gretter,matasso,falavi}@fbk.eu

Abstract

This paper describes the systems developed by FBK for the 2019 Spoken CALL Shared Task, that requires to automatically grade Swiss students, speaking German, that have to answer in English to German prompts. All answers are automatically transcribed, using an Automatic Speech Recognition (ASR) system, and labelled as *accept* or *reject* by a classifier. We developed an improved version of the baseline ASR system (made available by the organizers of the challenge), that has been used to produce better automatic transcriptions, from which a set of linguistic features are derived. Then, features vectors, computed at sentence level, are fed into a neural network based classifier that predicts the labels.

In this paper we describe the details of the developed ASR system, as well as the set of features used in the accept/reject classification task. We also discuss the impact of subsets of features on the final classification performance.

Index Terms: spoken language proficiency, non-native speech

1. Introduction

This work presents the FBK system developed for addressing the third Spoken Computer Assisted Language Learning (CALL) Shared Task[1] [1].

Generally, CALL is meant as the introduction of computers in the process of teaching/learning a second language. Specifically, scientific literature is rich in approaches for automated assessment of spoken language proficiency. Performance is directly dependent on ASR accuracy which, in turn, depends on the type of input, read or spontaneous, and on the speaker ages, adults or children (see [2] for an overview of spoken language technology for education). Automatic assessment of reading capabilities of L2 children was widely investigated in the past at both sentence level [3] and word level [4]. More recently, the scientific community has started to address automatic assessment of more complex spoken tasks, requiring more general communication capabilities by L2 learners. The AZELLA data set [5], developed by Pearson, includes $1,500$ spoken tests, each double graded by human professionals, from a variety of tasks. The work in [6] describes a latent semantic analysis (LSA) based approach for scoring the proficiency of the AZELLA test set, while [7] describes a system designed to automatically evaluate the communication skills of young English students. Features proposed for evaluation of pronunciation are described for instance in [8].

The winners of the second CALL shared task [9] use a deep neural network (DNN) model to accept or reject input utterances, while the work reported in [10] makes use of a support vector machine originally designed for scoring written texts.

Finally, it is worth mentioning that the recent end-to-end approach [11] (based on the usage of a bidirectional recurrent DNNs employing an attention model) performs better than the well known SpeechRater™system [12], for automatically scoring non-native spontaneous speech in the context of an online practice test for prospective takers of the Test Of English as a Foreign Language (TOEFL)[2].

In this paper, we first describe the ASR system employed to automatically transcribe the spoken answers of the students. The system uses the Kaldi toolkit [13] and, specifically, a discriminative training procedure based on lattice free Maximum Mutual Information (MMI [14]) and a training dataset comprising both children [15] and non-native speech [16] besides the provided in-domain material.

Then we discuss the set of features, derived from the automatic transcriptions of the answers of the students, used as input to a neural network (NN) based classifier which accepts or rejects the answers themselves. This feature set mainly originates from a previous work by us [17] aimed at automatically scoring the language proficiency of Italian students learning both English and German. Starting from the ASR output only, we computed 3 main types of features: *i)* features related to the coverage of a reference grammar provided together with the data of the challenge; *ii)* features derived from the errors computed through an edit distance between ASR output and the most similar sentence admitted by the reference grammar; *iii)* features measuring the similarity between the ASR output and some Language Models (LMs), built also with the automatic transcriptions of the training set. The impact of different subsets of features on the final performance of the system has been investigated and will be discussed.

The novelties proposed in this paper are as follows.

- An improved ASR system based on: *i)* the usage of specific speech including a combination of non-native and children speech; *ii)* an acoustic model employing time delay neural networks (TDNNs) trained in order to optimize a sequence level Bayes risk [14, 18, 19].
- The usage of features accounting for critical patterns in the automatic transcription of a given input utterance. For example, greetings at the sentence boundary (e.g. the word "please") are not relevant for the purpose of utterance classification, while either presence or lack of certain prepositions (e.g. "in", "at", etc) can significantly affect the final decision (i.e. *accept* or *reject* the utterance). To extract these features we propose to train an error model capable of weighting the edit errors between the recognized string and some reference valid answers (see section 4.1.1 for the details).

2. The third Spoken CALL Shared Task

The training portion of the Spoken CALL Shared Task is composed by a number of prompt-response pairs. Prompts are German written questions, associated with animation video clips (not included in the data) each showing an English native

[1] https://regulus.unige.ch/spokencallsharedtask_3rdedition

[2] TOEFL: https://www.ets.org/toefl

10.21437/SLaTE.2019-2

speaker asking a question, while responses are speech recordings of spoken utterances given in English by German-speaking Swiss teenagers. Each pair normally comes with additional information, namely: an unique *Id*, an orthographic transcription, the output of an ASR baseline, and two labels (each of them can be *correct* or *incorrect*) denoting the linguistic correctness of the responses in terms of *language* and *meaning*, respectively. Part of the training data, that have been semi-automatically labelled, contains also information that summarizes the labelling process of the corresponding utterances (field *Trace*). The task of the participants consists in scoring each test utterance with *accept* or *reject*; a response should be accepted only when both labels mentioned above result to be *correct*. Finally, a *reference grammar* is also provided which lists a number of possible "admitted" written replies for each given prompt. In the past editions of the challenge, the organizers produced two training sets and two evaluation sets.

For this work, we used the two training sets of the past editions as a unique training set (hereinafter denoted as *TrainingSet*), the test set of the first challenge as development set (*DevSet*), the test set of the second edition as evaluation set (*EvalSet*). Some statistics of these data sets are shown in Table 1. TrainingSet was used to train/adapt both acoustic models (AMs) and language models (LMs), to train error models and, of course, to train NN classifiers. *DevSet* and *EvalSet* were used to determine hyper-parameters of the whole system (e.g. LM weight, NN learning rate, number of NN learning iterations, etc) and to measure the corresponding performance, respectively. The *TestSet* of the challenge was blindly classified. Finally, note that the field *Trace*, introduced in the training set of the second challenge to keep track of how responses were labelled, and somehow denoting the difficulty of the labelling process, was not used in this work.

Id	# of Utterances	Source
DevSet	995	scst1 Test
EvalSet	1000	scst2 Test
TrainingSet	11919	scst1 Train + scst2 Train
TestSet	1000	scst3 Test

Table 1: *Data sets of the challenge used in this paper.*

3. ASR system

The adopted ASR system is built upon the baseline ASR released by the challenge organizers [20]; the acoustic model is trained according to a TDNN architecture while the language model is inherited from the baseline system.

3.1. Acoustic model

We have prepared new models adopting a popular Kaldi recipe, that features:

- a GMM triphone model trained using standard MFCC acoustic applying linear discriminative analysis (LDA), maximum likelihood linear transformation (MLLT), feature space maximum likelihood linear regression (fMLLR), and speaker adaptive training (SAT).
- the usage of i-vectors (of size 100) that are stacked to 40 MFCCs;
- a TDNN is trained using the lattice-free maximum mutual information approach [14]; the net is directly trained with a sequence-level objective function. MMI runs on GPUs and is implemented without using word lattices by applying a full forward-backward step on a decoding graph derived from a phone n-gram language model. A reduced frame rate is adopted and, therefore, a different HMM topology is derived.

The recognition of non-native speech, especially in the framework of multilingual speech recognition, is a well-investigated problem. Past research has tried to model the pronunciation errors of non-native speakers [21] both by using non-native pronunciation lexicons [22, 23, 24] or by adapting acoustic models with either native data and non native data [25, 26, 27, 28].

For the recognition of non-native speech, we demonstrated in [29] the effectiveness of adapting a multilingual deep neural network (DNN) trained on recordings of native speakers to children between 9 and 14 years old.

However, in this work acoustic model training is performed adding to the official in-domain training material (see Section 2 the following datasets:

• **PF-STAR** [15], specifically the recordings of read English speech spoken by German children (about 3.5h);

• **ISLE**, the Interactive Spoken Language Education corpus [16], consists of 11484 utterances recorded by intermediate-level German and Italian learners of English (about 18h).

3.2. Language models for ASR

Our first system uses the 3-gram stochastic language model provided by the organizers (see [20] for details). We have also trained a new language model using the manual transcriptions of the full TrainingSet and obtained a second ASR system that performs slightly better in terms of WER.

3.3. ASR performance

Table 2 reports the main features of the adopted ASR system and the resulting WERs on the EvalSet.

	AM	LM	WER
AsrV1	TDNN, LF-MMI, train on TrainingSet + PF-STAR + ISLE	3-grams scst1	7.6
AsrV2	TDNN, LF-MMI, train on TrainingSet + PF-STAR + ISLE	3-grams scst1+2	7.5

Table 2: *WER results on the EvalSet, provided by our ASR systems.*

4. Classification system

Feed-forward neural networks are used to classify features extracted from the automatically recognised utterances and described in section 4.1. The NN architecture and the related hyper-parameters have been optimized on the $DevSet$ defined in section 2.

As a results all NNs have three hidden layers and an output layer with four nodes, associated to the scores *correct* and *incorrect* of the proficiency indicators *language* and *meaning* as mentioned above.

The number of nodes of the hidden layers depends on the experiment (see section 5), but normally is equal to the number of input features. The loss function is the categorical cross-entropy. The activation function of all nodes is $ReLU$, while parameter estimation is done using Stochastic Gradient Descent (SGD) with Adaptive Gradient ($AdaGrad$). The learning rate is set to 0.05.

Table 3 shows the 4 possible outputs and the resulting final label (*accept* or *reject*).

language	meaning	class	judgement
incorrect	incorrect	0	reject
incorrect	correct	1	reject
correct	incorrect	2	reject
correct	correct	3	accept

Table 3: *Classifier output providing the class hypothesis associated to the two indicators* language *and* meaning*, and corresponding final judgement.*

4.1. Classification features

Classification features are obtained using both: *i)* language models trained on different corpora, including the automatic transcriptions of the training set and *ii)* the reference grammar provided by the organizers of the challenge. This one consists of a list of "correct" answers associated to each possible prompt.

In total we trained 12 stochastic LMs, computing 4 n-grams models ($1 \leq n \leq 4$) on the 3 text corpora that follows:
• **Generic**: around 3 millions of words belonging to transcriptions of English TED talks[3];
• **TrainRejRec**: ASR outputs, bounded by labels *_start_* and *_end_*, corresponding to the *incorrect* utterances of TrainingSet;
• **TrainAccRec**: ASR outputs, bounded by labels *_start_* and *_end_*, corresponding to the *correct* utterances of TrainingSet.

4.1.1. Error model

We propose to use an error model that allows to distinguish between critical and non critical errors in the ASR output of an utterance. This approach allows to avoid some text pre-processing, e.g. removing greeting words at the beginning of the responses as proposed in some of previous works (e.g. [30, 20]), to better match sentences in the reference grammar.

$\mathcal{C}(p,E,\overline{w}_p)$ / $\mathcal{E}(p,E,\overline{w}_p)$	$\mathcal{C}(p,E,\overline{w}_p)$ / $\mathcal{E}(p,E,\overline{w}_p)$
-1022 / 6-I-please	49 / 6-S-menu-card
-996 / 2-S-should-would	37 / 4-I-to
-739 / 8-I-please	32 / 5-S-on-at
-690 / 7-I-please	32 / 5-D-at
-678 / 2-S-need-want	24 / 8-S-cards-card
-599 / 1-I-no	24 / 7-S-a-credit
-515 / 5-I-please	24 / 7-D-credit
-490 / 9-I-please	24 / 6-D-the
-443 / 1-S-could-can	24 / 4-S-pay-buy
-392 / 4-S-one-a	24 / 2-I-don't

Table 4: *Most common edit errors ($\mathcal{E}$) found in* TrainingSet*. Negative values of corresponding counters ($\mathcal{C}$) indicate non critical errors, positive values indicate critical errors.*

To train the error model, each ASR output $W^k = w_1^k, \ldots, w_I^k$ in the TrainingSet ($1 \leq k \leq K$, being K the number of training utterances) is aligned with all the answers in the reference grammar corresponding to the related prompt; only the alignments exhibiting edit distance 1 (to account for more errors, more complex models could be considered in the future) are retained. The corresponding *edit error* $\mathcal{E}$ is represented by a triple $\mathcal{E}(p, E, \overline{w}_p)$, where p is its position inside the transcription, E is the type of error (i.e. substitution, insertion or deletion) and $\overline{w}_p$ are the words involved, and contributes to either increase or decrease (depending if the utterance k was labelled incorrect or correct, respectively) a corresponding counter $\mathcal{C}(p, E, \overline{w}_p)$. Of course, each edit error $\mathcal{E}(p, E, \overline{w}_p)$ can get contributions (positive or negative) from different sentences k in the TrainingSet. Table 4 reports the list of the most common critical (positive counter values) and not critical (negative counter values) errors. The rationale behind this approach is that, for instance, the deletion of some syntactically important words (e.g. deletion of preposition "at" in position 5, i.e. *"5-D-at"* in the Table) should lead to an error while the substitution between equivalent words (e.g. "should" replaced by "would" in position 2, i.e. *"2-S-should-would"* in the Table) or the insertion of the word "please" (i.e. *"6-I-please"*) should lead to accept. In this way the pre-processing step proposed in [30] can be substituted by a data driven approach where the distinction between crucial and not crucial errors can be hopefully learnt from training data.

From each training/test sentence we extract the following set of features:

- **standard**, 4 features, namely: number of automatically recognized words, number of content words, number of out-of-vocabulary (OOV) words, percentage of OOV;
- **reference**, 5 features computed using the reference grammar and the edit error, specifically: *i)* a binary value (1/0) if the sentence is or not admitted by the reference grammar; *ii)* a binary value (1/0) if the sentence has, or not, edit distance ≤ 1 from at least one of the admitted sentences in the grammar; *iii)* the minimum edit distance; *iv)* the edit error counter $\mathcal{C}$ having the highest absolute value; *v)* a smoothing function of the edit error counter $\mathcal{C}$, i.e.
$$\begin{array}{ll} +1 + 3 \times \log(\mathcal{C}) & \text{if } \mathcal{C} > 0 \\ -1 - 3 \times \log(-\mathcal{C}) & \text{if } \mathcal{C} < 0 \\ 0.0 & \text{if } \mathcal{C} = 0 \end{array}$$
- **LMs**, features computed by means of the selected LMs. The maximum number of this type of features is $5 \times 12 = 60$, when all the 12 LMs are selected. In fact, for each LM we extract the following 5 features, inspired by works described in [31, 32, 12, 9, 10]): *i)* $\frac{\log(P)}{N_W}$, that is, the average log-probability of the sentence, *ii)* $\frac{\log(P_{OOV})}{N_{OOV}}$, that is, the average contribution of OOV words to the log-probability of the sentence, *iii)* $\frac{\log(P) - log(P_{OOV})}{N_W}$, that is, the average log-difference between the two above probabilities, *iv)* $N_W - N_{bo}$, where N_{bo} is the number of back-offs applied by the LM to the input sentence (this difference is related to the frequency of n-grams in the sentence that have also been observed in the training set), *v)* N_{OOV}, the number of OOVs in the sentence. If word counts N_W or N_{OOV} are equal to zero (i.e. both P and P_{OOV} are not defined), the corresponding average log-probabilities are replaced by -1.

5. Experiments and conclusions

Several classification experiments were done before the challenge took place, considering the metric D_{full}. Concerning the 3 hidden layers forward NN topology, four configurations were tested: flat (number of hidden units on each layer equal to the number of features, NF), dflat (double flat, $NF \times 2$), hflat (half flat, $NF \times 0.5$) and pyr (pyramidal, with a decreasing number of hidden units: NF, $NF \times 0.8$, $NF \times 0.5$). Number of epochs was set to 100, 600, 1000, 3000. Figure 1 shows D_{full} results for the various cases, taking majority voting over 10 runs. Flat

[3] see https://www.ted.com for some details of this corpus.

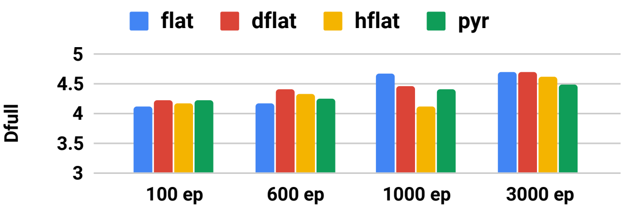

Figure 1: *D_{full} classification results on the DevSet, majority voting over10 runs, depending on NN topology and number of epochs.*

configuration and 1000/3000 training epochs appear to give the best results, on the DevSet.

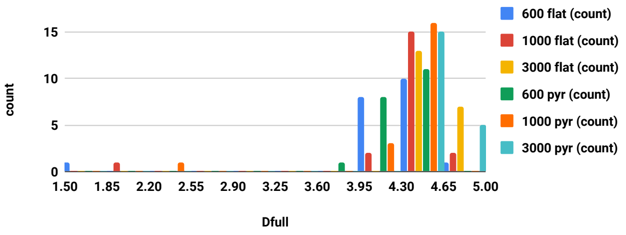

Figure 2: *Histogram of D_{full} classification results, 20 runs for each configuration; sparsness is mainly due to the random initialization of the NN.*

Majority voting was adopted after observing a high variability of the results, which mainly depends on the random initialization of the NN weights. Figure 2 reports some histograms of D_{full} classification results for 20 runs for each of 6 different NN configurations, showing a great variability. We tried some strategies to get stable results: given a topology, we took the 3/5/10 runs leading to the best results on the DevSet, and apply a majority voting to assign each single utterance the accept/reject decision. Figure 3 reports D_{full} results for some NN configurations, comparing the average D_{full} value computed on 10 runs (without majority voting) with the 3/5/10 majority voting cases. In most of the cases, the best3 and best5 majority voting assure the best results.

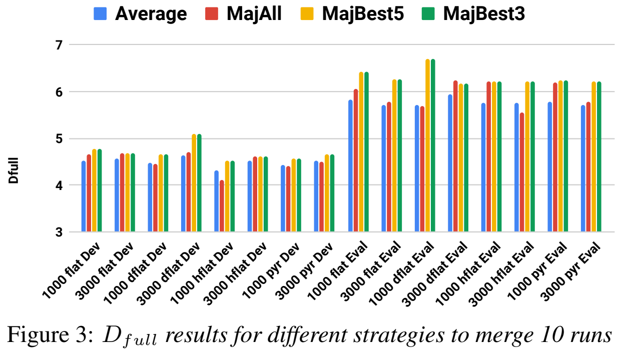

Figure 3: *D_{full} results for different strategies to merge 10 runs for each configuration, both on DevSet and EvalSet.* Maj *stands for* Majority voting.

Figure 4 reports D_{full} results for 3 different feature sets, and different NN configurations. *Final Features* contains all standard, all reference, and LMs features related to 4 LMs: TrainRejRec and TrainAccRec, 3-/4-grams; *More LMs* differs from *Final Features* because it contains 9 LMs: Generic, TrainRejRec, TrainAccRec, 2-/3-/4-grams; *No Edit Errors* differs from *Final Features* because it contains only the first of the 5 reference features. From this and other not reported experiments, it can be observed that adding the Generic LM does not help for this particular task, where most of the relevant information is already in the reference grammar; also unigrams and bigrams do not help. Furthermore, the comparison among *Final Features* and *No Edit Errors* allows to appreciate the important contribution of the error model.

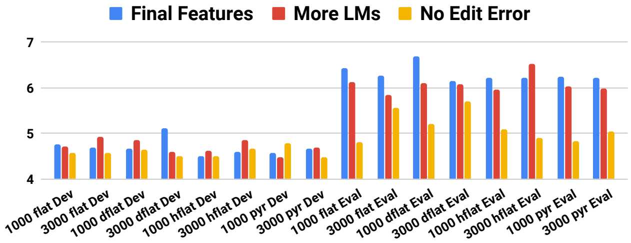

Figure 4: *D_{full} results for different feature set, best3 majority voting, both on DevSet and EvalSet.*

Concerning the official challenge, features used were always the *Final Features* described above and all the best3 majority voting were computed on the EvalSet from a set of 20 runs. These are the set-ups of the three submissions we made:

- **B3-A1-F-3** is the best3 majority voting, using AsrV1, flat topology and 3000 epochs (official submission GGG);
- **B6-B3-A1-FP-613** is the majority voting of 6 systems, each one being the best3 majority voting obtained with AsrV1, flat/pyr1 topology, 600/1000/3000 epochs (official submission HHH);
- **B12-B3-A12-FP-613** is the majority voting of 12 systems, each one being the best3 majority voting obtained with AsrV1/AsrV2, flat/pyr1 topology, 600/1000/3000 epochs (official submission III). The results obtained with the three submissions are reported in Table 5.

Id	DevSet	EvalSet	TestSet
B3-A1-F-3	4.833	6.343	**6.117**
B6-B3-A1-FP-613	4.912	6.247	**6.095**
B12-B3-A12-FP-613	5.115	5.169	**5.767**

Table 5: *D_{full} results for the official submissions to the 3rd challenge. TestSet is the official test set of the 3rd challenge.*

To conclude, we observe that the usage of hand-crafted features is effective for scoring the proficiency of the data of the challenge. In particular the introduction of (data-driven) features weighting the impact of certain word patterns on the final classification provided by the system allowed to significantly improve the performance.

We also noticed, as expected, that the final performance of the scoring system is highly dependent on the WER of the ASR system employed.

Finally, we have coped with the problem of local minima in NN classifiers by applying majority voting over a large number of different network initialization (this is possible due to the negligible training time of the NNs). In the future we will also try to adopt more complex network architectures as well as more effective initialization methods, such as the ones based on restricted Boltzman machines and contrastive divergence optimization [33].

6. 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).* Portorož, Slovenia: European Language Resources Association (ELRA), May 2016, pp. 237–244.

[2] M. Eskenazi, "An overview of spoken language technology for education. speech communication," *Speech Communication*, vol. 51, no. 10, pp. 2862–2873, 2009.

[3] K. Zechner, J. Sabatini, and L. Chen, "Automatic scoring of childrens read-aloud text passages and word lists," in *Proc. of the Fourth Workshop on Innovative Use of NLP for Building Educational Applications*, 2009.

[4] J. Tepperman, M. Black, P. Price, S. Lee, A. Kazemzadeh, M. Gerosa, M. Heritage, A. Alwan, and S. Narayanan, "A bayesian network classifier for word-level reading assessment," in *Proc. of ICSLP*, 2007.

[5] J. Cheng, Y. Zhao-DAntilio, X. Chen, , and J. Bernstein, "Automatic spoken assessment of young english language learners," in *Proc. of the Ninth Workshop on Innovative Use of NLP for Building Educational Applications*, 2014.

[6] M. Angeliki and C. J., "Using Deep Neural Networks to improve proficiency assessment for children English language learners," in *Proc. of Interspeech*, 2014, pp. 1468–1472.

[7] K. Evanini and X. Wang, "Automated speech scoring for nonnative middle school students with multiple task types," in *Proc. of Interspeech*, 2013, pp. 2435–2439.

[8] H. Kibishi, K. Hirabayashi, and S. Nakagawa, "A statistical method of evaluating the pronunciation proficiency/intelligibility of english presentations by japanese speakers," *ReCALL*, vol. 27, no. 1, pp. 58–83, 2015.

[9] Y. R. Oh, H.-B. Jeon, H. J. Song, B. O. Kang, Y.-K. Lee, J. Park, and Y.-K. Lee, "Deep-learning based Automatic Spontaneous Speech Assessment in a Data-Driven Approach for the 2017 SLaTE CALL Shared Challenge," in *Proc. of SlaTe*, Stockholm, Sweden, 2017, pp. 103–108.

[10] K. Evanini, M. Mulholland, E. Tsuprun, and Y. Qian, "Using an Automated Content Scoring System for Spoken CALL Responses: The ETS submission for the Spoken CALL Challenge," in *Proc. of SlaTe*, Stockholm, Sweden, 2017, pp. 97–102.

[11] L. Chen, J. Tao, S. Ghaffarzadegan, and Y. Qian, "End-to-end neural network based automated speech scoring," in *Proc. of ICASSP*, Calgary, Canada, 2018, pp. 6234–6238.

[12] K. Zechner, D. Higgins, X. Xi, and D. Williamson, "Automatic scoring of nonnative spontaneous speech in tests of spoken English," *Speech Communication*, vol. 51, no. 10, pp. 883–895, 2009.

[13] D. Povey, A. Ghoshal, G. Boulianne, L. Burget, O. Glembek, N. Goel, M. Hannemann, P. Motlicek, Y. Qian, P. Schwarz, J. Silovsky, G. Stemmer, and K. Veselý, "The Kaldi Speech Recognition Toolkit," in *Proc. of IEEE ASRU Workshop*, Hawaii (US), December 2011.

[14] D. Povey, V. Peddinti, D. Galvez, P. Ghahremani, V. Manohar, X. Na, Y. Wang, and S. Khudanpur, "Purely sequence-trained neural networks for ASR based on lattice-free MMI," in *Proc. of Interspeech*, 2016, pp. 2751–2755.

[15] 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. of Eurospeech*, 2005, pp. 2761–2764.

[16] W. Menzel, E. Atwell, P. Bonaventura, D. Herron, P. Howarth, R. Morton, and C. Souter, "The ISLE corpus of non-native spoken English," in *Proceedings of LREC*, 2000, pp. 957–964.

[17] M. Matassoni, K. Allgaier, S. Tchistiakova, and D. Falavigna, "Automatic assessment of spoken language proficiency of non-native children," in *Proc. of ICASSP*, Brighton, UK, 2019.

[18] V. Peddinti, Y. Wang, D. Povey, and S. Khudanpur, "Low latency acoustic modeling using temporal convolution and lstms," *IEEE Signal Process. Lett.*, vol. 25, no. 3, pp. 373–377, 2018.

[19] H. Hadian, H. Sameti, D. Povey, and S. Khudanpur, "End-to-end speech recognition using lattice-free MMI," in *Interspeech*. ISCA, 2018, pp. 12–16.

[20] M. Qian, X. Wei, P. Jancovic, and M. Russell, "The university of birmingham 2017 slate call shared task systems," in *Proceedings of the 7th ISCA Workshop on Speech and Language Technology in Education*, O. Engwall, J. Lopes, and I Leite, Eds. ISCA, 8 2017.

[21] G. Bouselmi, D. Fohr, I. Illina, and J. P. Haton, "Multilingual non-native speech recognition using phonetic confusion-based acoustic model modification and graphemic constraints," in *Proc. of ICSLP*, 2006, pp. 109–112.

[22] Z. Wang, T. Schultz, and A. Waibel, "Comparison of acoustic model adaptation techniques on non-native speech," in *Proc. of ICASSP*, 2003, pp. 540–543.

[23] Y. R. Oh, J. S. Yoon, and H. K. Kim, "Adaptation based on pronunciation variability analysis for non native speech recognition," in *Proc. of ICASSP*, 2006, pp. 137–140.

[24] H. Strik, K. Truong, F. de Wet, and C. Cucchiarini, "Comparing different approaches for automatic pronunciation error detection," *Speech Communication*, vol. 51, no. 10, pp. 845–852, 2009.

[25] R. Duan, T. Kawahara, M. Dantsuji, and J. Zhang, "Articulatory modeling for pronunciation error detection without non-native training data based on dnn transfer learning," *IEICE Transactions on Information and Systems*, vol. E100.D, no. 9, pp. 2174–2182, 2017.

[26] W. Li, S. M. Siniscalchi, N. F. Chen, and C. Lee, "Improving non-native mispronunciation detection and enriching diagnostic feedback with DNN-based speech attribute modeling," in *Proc. of ICASSP*, 2016, pp. 6135–6139.

[27] A. Lee and J. Glass, "Mispronunciation detection without nonnative training data," in *Proc. of Interspeech*, 2015, pp. 643–647.

[28] A. Das and M. Hasegawa-Johnson, "Cross-lingual transfer learning during supervised training in low resource scenarios," *Proc. of Interspeech*, pp. 3531–3535, 2015.

[29] M. Matassoni, R. Gretter, D. Falavigna, and D. Giuliani, "Non-native children speech recognition through transfer learning," in *Proc. of ICASSP*, Calgary, Canada, 2018, pp. 6229–6233.

[30] H. Nguyen, L. Chen, R. Prieto, C. Wang, and Y. Liu, "Liulishuos system for the spoken call shared task 2018," *Proc. Interspeech 2018*, pp. 2364–2368, 2018.

[31] K. Sakaguchi, M. Heilman, and N. Madnani, "Effective feature integration for automated short answer scoring," in *Proc. of NAACL*, Denver (CO), USA, 2015, pp. 1049–1054.

[32] S. Srihari, R. Srihari, P. Babu, and H. Srinivasan, "On the automatic scoring of handwritten essays," in *Proc. of IJCAI*, Hyderabad, India, 2007, pp. 2880–2884.

[33] G. Hinton, L. Deng, D. Yu, and Y. Wang, "Deep Neural Networks for Acoustic Modeling in Speech Recognition," *IEEE Signal Processing Magazine*, vol. 9, no. 3, pp. 82–97, 2012.