Unusable Spoken Response Detection with BLSTM Neural Networks

Zhaoheng Ni[1], Rutuja Ubale[2], Yao Qian[2], Michael Mandel[13], Su-Youn Yoon[2], Abhinav Misra[2], David Suendermann-Oeft[2]

[1]Computer Science Program, The Graduate Center, City University of New York
[2]Educational Testing Service Research, USA
[3]Computer and Information Science, Brooklyn College, City University of New York

zni@gradcenter.cuny.edu, rubale@ets.org, yqian@ets.org, mim@mr-pc.org, syoon@ets.org, amisra001@ets.org, suendermann-oeft@ets.org

Abstract

Voice biometrics has been applied to enhance the security of spoken language proficiency tests and ensure valid test scores by detecting fraudulent activity. These methods can, however, be triggered by certain distortions, including background noise and adjacent test-takers, resulting in false positive alarms. In this paper, a two-layer bi-directional LSTM RNN model is employed to detect these distorted (unusable) responses and a sub-sampling method is applied to reduce the difficulties of model training caused by very long input sequence and imbalanced training data. The system is evaluated on a corpus that was collected from an assessment of English language proficiency around the world. Results show that our approach significantly outperforms two baselines: a Gaussian mixture model (GMM) classifying frame-level features and an AdaBoost classifier operating on i-vectors. Our system's F-score in unusable response detection is 0.60 compared to 0.43 and 0.49 for the two baseline systems.

Index Terms: voice biometrics, unusable response detection, bi-directional LSTM RNN paralinguistics

1. Introduction

In automated test assessment systems, a speech biometric subsystem is an essential component to verify if the test participant is the registered user. The state-of-the-art approach for speaker verification is to extract i-vector features [1] from two utterances and calculate the distance between them. If the distance is above a threshold, then the speakers of the two utterances are considered to be different. This approach is complicated by the fact that the spoken responses can be collected from different environments (room size, microphone type, background noise). Furthermore, in some recordings, the signal is dominated by the background noise or there is no speech in the audio at all. Another distortion comes from technical recording issues such as clipping, which greatly reduces the quality and intelligibility of the speech. In all cases, the spoken responses cannot be analyzed by the speech biometric system. To improve speaker verification performance, then, we need to design a model to filter out the unusable responses automatically. This model can also be applied to evaluate the quality of the audio recording before the test starts, helping the user make adjustments to avoid recording issues altogether.

In previous work, Higgins et al. [2] developed a filtering model to filter non-scorable responses caused by technical problems before sending recordings to a speech rater system. They used a regression model based on four features: the number of distinct words in the speech recognition output, the average speech recognizer confidence score, the average power of the speech signal, and the mean absolute deviation of the speech signal power. The model achieved 90% accuracy, but 38.5% F-score because the data are imbalanced: only around 1% of the speech was non-scorable. Yoon et al. [3] extracted features from automatic speech recognition results and signal processing and trained a decision tree model on the features to distinguish between non-scorable spoken responses and normal spoken responses. They achieved 96.8% accuracy and 59.4% F-score on spoken responses of the international English proficiency test dataset for the test participants in India. The non-scorable data comprise around 30% of the spoken responses. In our task, the class distribution is as imbalanced as that in [2], which makes it challenging to fine-tune the classification model. The spoken responses are recorded with different environments (microphone type, room geometry, noise type) and by speakers from different countries, which makes the task more challenging.

Many research show the imbalanced class distribution affects the performance of the classification models [4][5][6]. Liu et al. [7] trained multiple classifiers on subsets of the majority class respectively and cascade the models. Training the models on small proportions of the data may fail to cover all patterns distributed in the samples, especially for the deep neural networks which require large-scale data. Yap et al. [8] show that over-sampling the minority class samples helps improve the classification performance. Based on the idea, we investigate the methods to enlarge the size of unusable speech samples.

Recently, Deep Neural Networks with Long Short-Term Memory (LSTM) structure has been widely applied in many applications of machine learning and achieved state-of-the-art performances [9][10][11][12]. The model is designed for processing time-series data. While few papers have made use of the temporal information of the speech in this task, as we do here, because sequential patterns in the spoken responses can help us distinguish unusable speech from normal speech. The gates and memory cells inside the LSTM structure prevent vanishing gradient throughout the back propagation [13], however, it's still difficult when the input is extremely long. In this paper, we design a bidirectional LSTM neural network model [14] for unusable spoken response detection. We avoid using ASR features compared with [3] since they depend on the performance of the ASR system and they are also not suitable when applying the detection model as the front-end in the testing software. Instead, we extract the features from the spoken response itself using signal processing methods and put them in sequential order. Then we train a BLSTM model on the extracted features. To overcome the long-input problem and the data imbalance problem, we design a sub-sampling method to enlarge the number of unusable speech samples in the training data and a unanimous-voting method to improve the robustness of the predictions.

2. Data

We used a large collection of spoken responses from an international English proficiency assessment. The assessment prompts speakers to provide responses lasting between 45 and 60 seconds each. All items elicit spontaneous, unconstrained, natural speech.

The responses were collected from a large number of different test centers, and some spoken responses had serious technical difficulties that obscured the content of the responses despite various efforts to control the quality of the recordings. Trained human raters labeled these responses as unusable.

Analyzing these unusable responses in our dataset revealed four different types:

- No Response: Total Silence. The silence in this category is what you would hear if the microphone was not plugged in. No ambient noise can be heard. (Note: This is not the same as when the candidate simply chooses not to speak and noises such as breathing, coughing, or the occasional background sound can be heard).
- Constant Noise: Responses with constant noise in the recording that obscures the candidate response (buzzing, clicking, crackling, static, or other mechanical noise).
- Distorted recording: Spoken response is too distorted to evaluate fairly. This group includes responses with a fast playback, slow playback, or an over-amplified and distorted spoken response not covered by the "Noise Constant" category.
- Missing Samples: Parts of the audio are missing and the test taker's voice often cuts in and out.

We evaluate our system by two speech detection tasks: a) distinguishing between speech and non-speech and b) between usable speech and unusable speech.The distribution of normal speech and unusable speech is highly unbalanced. In non-speech responses, there are only "No Response" and "Constant Noises". The unusable spoken responses cover all types described above, which is more challenging. The data partition is shown in Table 1. The numbers of spoken responses for the two classes in these two tasks are designed according to the distributions in the real data set. The response is labeled as non-speech by human raters if there is no speaker talking in the whole audio. The response is labeled as normal speech if the score given by the raters is greater than zero.

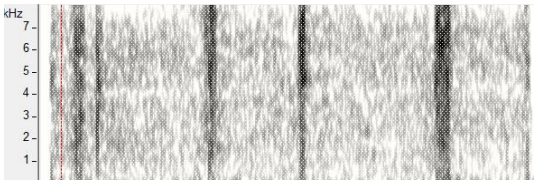

Figure 1: *Non-speech due to microphone malfunction*

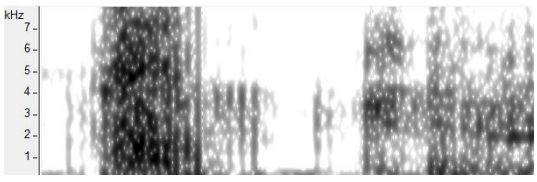

Figure 2: *Unusable speech due to clipping*

Table 1: *Data Partition for Two Detection Tasks*

Task	Training	Test
speech/non-speech	40,000/400	40,000/400
usable/unusable	43,027/579	43,822/580

Figure 1 and Figure 2 show examples of non-speech and unusable spoken responses, respectively.

3. Method

3.1. BLSTM Model

Recently, LSTM Recurrent Neural Networks (RNNs) have been successfully applied in many machine learning fields (e.g., speech recognition, fMRI classification), especially when dealing with time-series data. This model is designed to address the vanishing gradient problem observed in traditional RNNs [13]. Given an input sequence $X = (x_1, x_2, x_3, \ldots, x_T)$, the states of an LSTM evolving according to:

$$\begin{pmatrix} \tilde{f}_t \\ \tilde{i}_t \\ \tilde{o}_t \\ \tilde{g}_t \end{pmatrix} = W_h h_{t-1} + W_x x_t + b, \tag{1}$$

$$c_t = \sigma(\tilde{f}_t) \odot c_{t-1} + \sigma(\tilde{i}_t) \odot tanh(\tilde{g}_t), \tag{2}$$

$$h_t = \sigma(\tilde{o}_t) \odot tanh(c_t), \tag{3}$$

where $W_h \in \mathbb{R}^{4d_h \times d_h}$, $W_x \in \mathbb{R}^{4d_x \times d_x}$, σ is the logistic function, and the $\odot$ operation is the Hadamard product.

The drawback of the LSTM model is that it can only make use of past information. One solution to this problem is the bidirectional LSTM model (BLSTM) [9][10]. In a BLSTM, two separate LSTM layers operate on the same input sequence, one in the forward direction and the other in the backward direction. In this way, both past and future context can be utilized to improve the performance.

3.2. Sub-sampling Data

Unusable spoken response detection can be regarded as a sequence-to-tag task, in which the binary classification is performed based on the given acoustic feature sequence. BLSTM is powerful for time series prediction and has been successfully applied to solve a wide range of machine learning problems with sequence data. However, when we apply BLSTM for our task, we need to solve two problems:

1) Very long sequence. The spoken responses last between 45 to 60 seconds. The length of the sequence in terms of the number of frames could be 4,500 to 6,000 if short time Fourier transform (STFT) with 10 ms windows shift is employed to acoustic feature extraction. Although BLSTM is capable of learning and remembering long sequences of inputs, it can still be a challenge for BLSTM if the sequence-to-tag mapping with only one output while very long input sequences are used in the task. The challenge comes from model converging too slowly or vanishing gradients which may result in an unlearnable model.

2) Imbalanced or skewed class distribution. The percentage of unusable spoken responses generally ranges from 1% to 5% in the training data which are randomly collected from an international English proficiency assessment. Zhu et al.[6] show that the class imbalance increases the difficulty of training the cost-sensitive neural networks. Training the neural networks tends to be overwhelmed by the class that makes up a larger proportion

of the training data. It is essential to balance the class distribution.

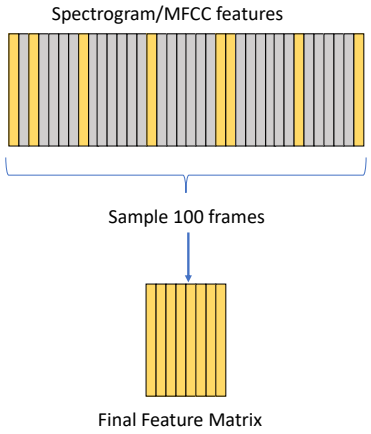

Figure 3: *Sub-sampling features*

Du et al.[15] employ a random sub-sampling approach to 3D convolutional neural networks based video action detection. The system picks 16 frames randomly distributed between the beginning and the end of the video, then concatenates these frames to form the final feature vector sequence. Their results show that sub-sampled features preserve the information in the video. Inspired by their promising results, we extend the sub-sampling idea to BLSTM-based unusable spoken responses detection. The response is finally represented by an N*M feature matrix, where N is the number of sub-sampled frames and M is the dimension of STFT-based feature vector. We set N to be 100. Three different sub-sampling strategies are investigated for addressing the problem of very long input sequence, as shown in Figure 3:

- Extract 100 contiguous frames in the middle of the audio
- Extract 100 random frames between the beginning and the end and concatenate these frames together
- Split the audio into 5/10/20 equal segments, concatenate the first 20/10/5 contiguous frames from each segment

To overcome the data imbalance problem, we increase the number of samples for every unusable response while we only extract one feature matrix from every usable response.

Algorithm 1 The procedure to balance class distribution

Let X be the MFCC features of the unusable speech. The shape of X is $T \times M$, where T is the total number of frames, and M is the dimension of feature vector.
procedure UP-SAMPLE(X) ▷ Extract sub-sampled features
  $training \leftarrow \{\}$
  for $itr \in 1:100$
    $sample \leftarrow \{\}$
    for $i \in 1:N$
      $rand \leftarrow random(1, T)$
      $sample \leftarrow sample \cup X[rand, 1:M]$
    end for
    Sort $sample$ in sequential order
    $training \leftarrow training \cup sample$
  end for

The procedure is described as shown in Algorithm 1. Each sample is a $N * M$ feature matrix. We generate 100 such samples from every unusable spoken response and only generate one sample from every usable speech. For contiguous frames and equal-segment frames, we shift the starting point with window size as 1 and extract 100/5/10/20 for each segment. By this way, the number of unusable speech feature samples is the same as the number of usable speech feature samples. We split the samples into 70% as the training set and 30% as the validation set and feed them to the BLSTM models.

3.3. Unanimous-voting Method

For the testing procedure, we extract 5 random samples of 100 frames from each spoken response. For the other sub-sampling methods, the 5 samples are extracted with window shift 1 in every segment. Then we predict 5 labels for the 5 samples. The formula is defined as followed:

$$Label = \begin{cases} 0, & \text{if } P_i \geq threshold, \forall i \in [1,5] \\ 1, & \text{otherwise} \end{cases} \quad (4)$$

If all the samples predictions are above the threshold, the final prediction is set to be unusable speech, otherwise, the prediction is usable speech. We fine tune the threshold on the validation data for each sub-sampling method.

4. Experiments

We extract spectrogram and MFCC features using the LibROSA Python library [16]. The length of the FFT window for the spectrogram features is 512 samples with 256 overlap, thus the dimension of the frequency domain vector is 257. The length of the FFT window for MFCC features is 2048 and the dimension of MFCC features is 39. The overlap size between frames is 512.

We train the BLSTM models using Keras Toolkit [17] with TensorFlow [18] as the backend. All models have two BLSTM layers with 32 neural units and one fully-connected layer with one neural unit. The activation function after the output layer is set to be a sigmoid function. We use RMSProp optimizer with a learning rate of 0.001. We train the models with 80 epochs in total, saving the weights of the model with the highest validation accuracy.

We evaluate our system on the two tasks described in Section 2. We apply our sub-sampling method to the training data and train the BLSTM model on the generated features. For evaluation, there is no way to only enlarge the non-speech or unusable speech samples in the test data since we don't know the ground truth. We apply the unanimous-voting method on each spoken response and get the final prediction. We compare our system with the following baselines.

4.1. Gaussian Mixture Model Baseline

As a comparison for the BLSTM model, we use a likelihood test between two GMMs. First we extract MFCC features using Kaldi's feature extraction functions [19]. Then we train two GMM models, one on the non-speech/distorted speech and the other on the usable speech. The prediction is at the frame level. In the testing process, we calculate the average log-likelihood

of all the frames as follows:

$$L(usable) = \frac{1}{T}\sum_{i=1}^{T}\log \sum_{g \in usable} P(X_i \mid Y = g) \quad (5)$$

$$L(unusable) = \frac{1}{T}\sum_{i=1}^{T}\log \sum_{g \in unusable} P(X_i \mid Y = g) \quad (6)$$

$$ratio = \frac{L(unusable)}{L(usable)} \quad (7)$$

Where g indexes Gaussians in the mixtures, X_i is the MFCC feature for frame i, T is the number of frames in the spoken response. We set a threshold t, the response is labeled as unusable if $ratio > t$, otherwise as usable.

4.2. i-vector-based Baseline

An i-vector, which is a compact, low dimensional vector representation of an utterance, is a factor analysis based approach to modeling speech in a text-independent manner [1]. This approach addresses the problem caused by the variable length of spoken utterances. It is the state-of-the-art front-end for speaker recognition and language recognition. To extract the i-vector, a low-rank, rectangular T-matrix, which represents the total variability in the acoustic space, is estimated using the EM algorithm. A T-matrix estimated from 800 hours of speech from 8,700 test-takers, which was used to build a speaker recognition system [20], is applied here to extract i-vectors as the feature vectors for a second baseline system. Decision tree (DT), SVM, AdaBoost, and logistic regression (LR) classifiers were compared for classifying the i-vectors. Among those classifiers trained on the same training set and evaluated on the same test set as those used for GMM baseline, the AdaBoost classifier achieves the best performance, so we report its results as the baseline results for the i-vector-based approach.

5. Results and Analysis

We evaluate the systems' performance using precision, recall, and F-score metrics. The corresponding results are shown in Table 2 and Table 3, respectively. Table 2 shows that the GMM model achieved an F-score of 0.81 and the AdaBoost classifier using i-vectors achieves an F-score of 0.89. Comparing the features with different sub-sampling methods, we find the model trained on 100 random sub-sampled frames gets a higher F-score than the one trained on contiguous frames. The results make sense since the 100 random frames cover the information from the beginning to the end, while the contiguous frames only contain the information of one segment in the audio. We then investigated the performances of different sub-sampling methods for the MFCC features. The BLSTM model trained on random frames achieves an F-score of 0.90, which outperforms the conventional approaches, i.e., GMM baseline and i-vector-based baseline. The performance gap between our proposed approach and the i-vector-based approach is marginal, however.

The task of unusable speech detection is much more challenging than the non-speech detection task. We train our BLSTM model on random sub-sampled features. Table 3 shows that the BLSTM model achieves an F-score of 0.60, which is much higher than the GMM and i-vector baselines. We analyze some audio examples from the spoken responses and find that even though some spoken responses are distorted due to microphone issues when the human rater can still understand the speech, they will give it a non-zero score. The human experts will not label the speech as "distorted" unless it is so highly distorted that it is unintelligible.

[1] Continuous frames [2] Random frames [3] 5 × 20 frames [4] 10 × 10 frames [5] 20 × 5 frames [6] Random frames

Table 2: *Non-speech Detection Results*

Model	Feature	Precision	Recall	F-score
GMM	MFCC	0.76	0.89	0.81
AdaBoost	i-vector	0.92	0.87	0.89
BLSTM	Spectrogram (A)[1]	0.71	0.68	0.69
BLSTM	Spectrogram (B)[2]	0.91	0.81	0.86
BLSTM	MFCC (A)[3]	0.86	0.84	0.85
BLSTM	MFCC (B)[4]	0.63	0.78	0.70
BLSTM	MFCC (C)[5]	0.92	0.59	0.72
BLSTM	MFCC (D)[6]	0.90	0.90	0.90

Table 3: *Unusable Speech Detection Results*

Model	Feature	Precision	Recall	F-score
GMM	MFCC	0.51	0.38	0.43
AdaBoost	i-vector	0.59	0.41	0.49
BLSTM	MFCC[6]	0.61	0.58	0.60

One of motivations for using BLSTM-RNN with acoustic features inputs is to avoid ASR system for unusable spoken response detection. As a comparison, we investigate whether the number of recognized words from an ASR system can be used as a feature to distinguish between usable responses and unusable responses. Our ASR system [21] with i-Vector based speaker adaptation technology can deliver a WER of 18.5% on data used in this study. Table 4 shows the mean and standard deviation of speech, non-speech, usable and unusable response hypothesis. The statistics show that speech and non-speech is distinguishable, but it's difficult to distinguish between usable and unusable spoken responses by using the ASR hypothesis.

Table 4: *ASR Hypothesis Word Number Statistics*

Task	Data	Mean	STD
Speech/Non-speech	speech	99.22	29.74
	non-speech	6.34	8.80
Usuable/Unusable	usable	101.00	30.05
	unusuable	70.54	46.12

6. Conclusion

In this paper, we propose a BLSTM neural network to detect non-speech and unusable spoken responses to spoken language proficiency tests. This system can help filter out unusable responses and improve speaker verification performance. The model trained on sub-sampled MFCC features outperforms baseline models in both tasks. In the future, we would like to generalize our sub-sampling method to other time-series data classification tasks. The results show that there is still space to improve the unusable response classification result. One possible way is combining the DNN-based model with i-vector features to achieve a better performance (e.g. train a BLSTM-based model to extract better i-vectors).

7. References

[1] N. Dehak, P. J. Kenny, R. Dehak, P. Dumouchel, and P. Ouellet, "Front-end factor analysis for speaker verification," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 19, no. 4, pp. 788–798, 2011.

[2] D. Higgins, X. Xi, K. Zechner, and D. Williamson, "A three-stage approach to the automated scoring of spontaneous spoken responses," *Computer Speech & Language*, vol. 25, no. 2, pp. 282–306, 2011.

[3] S.-Y. Yoon, K. Evanini, and K. Zechner, "Non-scorable response detection for automated speaking proficiency assessment," in *Proceedings of the 6th Workshop on Innovative Use of NLP for Building Educational Applications*. Association for Computational Linguistics, 2011, pp. 152–160.

[4] N. Japkowicz and S. Stephen, "The class imbalance problem: A systematic study," *Intelligent data analysis*, vol. 6, no. 5, pp. 429–449, 2002.

[5] N. Japkowicz, "The class imbalance problem: Significance and strategies," in *Proc. of the Intl Conf. on Artificial Intelligence*, 2000.

[6] Z.-H. Zhou and X.-Y. Liu, "Training cost-sensitive neural networks with methods addressing the class imbalance problem," *IEEE Transactions on Knowledge and Data Engineering*, vol. 18, no. 1, pp. 63–77, 2006.

[7] X.-Y. Liu, J. Wu, and Z.-H. Zhou, "Exploratory undersampling for class-imbalance learning," *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, vol. 39, no. 2, pp. 539–550, 2009.

[8] B. W. Yap, K. A. Rani, H. A. A. Rahman, S. Fong, Z. Khairudin, and N. N. Abdullah, "An application of oversampling, undersampling, bagging and boosting in handling imbalanced datasets," in *Proceedings of the first international conference on advanced data and information engineering (DaEng-2013)*. Springer, 2014, pp. 13–22.

[9] A. Graves, S. Fernández, and J. Schmidhuber, "Bidirectional lstm networks for improved phoneme classification and recognition," in *International Conference on Artificial Neural Networks*. Springer, 2005, pp. 799–804.

[10] S. Zhang, D. Zheng, X. Hu, and M. Yang, "Bidirectional long short-term memory networks for relation classification," in *Proceedings of the 29th Pacific Asia Conference on Language, Information and Computation*, 2015, pp. 73–78.

[11] Z. C. Lipton, D. C. Kale, C. Elkan, and R. Wetzel, "Learning to diagnose with lstm recurrent neural networks," *arXiv preprint arXiv:1511.03677*, 2015.

[12] O. Vinyals, A. Toshev, S. Bengio, and D. Erhan, "Show and tell: A neural image caption generator," in *Computer Vision and Pattern Recognition (CVPR), 2015 IEEE Conference on*. IEEE, 2015, pp. 3156–3164.

[13] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural computation*, vol. 9, no. 8, pp. 1735–1780, 1997.

[14] A. Graves and J. Schmidhuber, "Framewise phoneme classification with bidirectional lstm and other neural network architectures," *Neural Networks*, vol. 18, no. 5-6, pp. 602–610, 2005.

[15] D. Tran, L. Bourdev, R. Fergus, L. Torresani, and M. Paluri, "Learning spatiotemporal features with 3d convolutional networks," in *Proceedings of the IEEE international conference on computer vision*, 2015, pp. 4489–4497.

[16] B. McFee, C. Raffel, D. Liang, D. P. Ellis, M. McVicar, E. Battenberg, and O. Nieto, "librosa: Audio and music signal analysis in python," in *Proceedings of the 14th python in science conference*, 2015, pp. 18–25.

[17] F. Chollet *et al.*, "Keras: The python deep learning library," 2015. [Online]. Available: https://keras.io

[18] M. Abadi, A. Agarwal, P. Barham, E. Brevdo, Z. Chen, C. Citro, G. S. Corrado, A. Davis, J. Dean, M. Devin *et al.*, "Tensorflow: Large-scale machine learning on heterogeneous distributed systems," *arXiv preprint arXiv:1603.04467*, 2016.

[19] D. Povey, A. Ghoshal, G. Boulianne, L. Burget, O. Glembek, N. Goel, M. Hannemann, P. Motlicek, Y. Qian, P. Schwarz *et al.*, "The kaldi speech recognition toolkit," in *IEEE 2011 workshop on automatic speech recognition and understanding*, no. EPFL-CONF-192584. IEEE Signal Processing Society, 2011.

[20] Y. Qian, J. Tao, D. Suendermann-Oeft, K. Evanini, A. V. Ivanov, and V. Ramanarayanan, "Noise and metadata sensitive bottleneck features for improving speaker recognition with non-native speech input," in *Proceedings of Interspeech*, 2016, pp. 3648–3652.

[21] Y. Qian, X. Wang, K. Evanini, and D. Suendermann-Oeft, "Self-adaptive DNN for improving spoken language proficiency assessment," *in Proc. of Interspeech*, 2016.