INTERSPEECH 2017
August 20–24, 2017, Stockholm, Sweden

Detection of Mispronunciations and Disfluencies in Children Reading Aloud

Jorge Proença[1,2], *Carla Lopes*[1,3], *Michael Tjalve*[4], *Andreas Stolcke*[5], *Sara Candeias*[6], *Fernando Perdigão*[1,2]

[1] Instituto de Telecomunicações, Coimbra, Portugal
[2] Department of Electrical and Computer Engineering, University of Coimbra, Portugal
[3] Polytechnic Institute of Leiria, Leiria, Portugal
[4] Microsoft & University of Washington, Seattle, WA, USA
[5] Microsoft AI and Research, Mountain View, CA, USA
[6] Microsoft, Digital Advisory Services, Lisbon, Portugal

{jproenca,calopes,fp}@co.it.pt, {michael.tjalve,andreas.stolcke,v-sacand}@microsoft.com

Abstract

To automatically evaluate the performance of children reading aloud or to follow a child's reading in reading tutor applications, different types of reading disfluencies and mispronunciations must be accounted for. In this work, we aim to detect most of these disfluencies in sentence and pseudoword reading. Detecting incorrectly pronounced words, and quantifying the quality of word pronunciations, is arguably the hardest task. We approach the challenge as a two-step process. First, a segmentation using task-specific lattices is performed, while detecting repetitions and false starts and providing candidate segments for words. Then, candidates are classified as mispronounced or not, using multiple features derived from likelihood ratios based on phone decoding and forced alignment, as well as additional meta-information about the word. Several classifiers were explored (linear fit, neural networks, support vector machines) and trained after a feature selection stage to avoid overfitting. Improved results are obtained using feature combination compared to using only the log likelihood ratio of the reference word (22% versus 27% miss rate at constant 5% false alarm rate).

Index Terms: children's speech, reading disfluencies, mispronunciation detection

1. Introduction

Reading aloud by children who are still learning how to read can present several problems that reflect their different levels of fluency. This oral reading fluency depends on speed, accuracy, consistency of pace and expressiveness [1]. Disfluencies and reading mistakes can vary from reading syllable by syllable to severe mispronunciations of a word, and present a significant challenge to automatic systems that aim to either evaluate a child's reading or to monitor their reading attempts (such as in automatic reading tutors).

There are several known methods to detect disfluencies, such as based on hidden Markov models (HMMs), maximum entropy models, conditional random fields [2] and classification and regression trees [3], though most of these past efforts focus on spontaneous speech. For read speech, there are differences in the types of events found, since different speaking styles vary in the production of disfluencies [4]. Certain works have targeted the detection of disfluencies in children's reading: Black et al. [5] target mostly sounding-outs of a word that can be whispered and use a grammar structure allowing partial words and silence or noise between phones; Duchateau et al. [6] use a phoneme-level lattice to allow false starts and partial pronunciations and a second unit to allow repetitions and deletions of words; Yilmaz et al. [7] added to a flexible decoding scheme the most common substitutions, deletions and insertions of phones in the language described by a phone confusion matrix; Li et al. [8] employed a context-free grammar with sentence words concurrent with other common words. However, most works focus on individual word reading tasks (with some exceptions in [7], [8]), whereas our work will focus on sentence and pseudoword reading. Some studies use the output of disfluency detection to provide an overall reading performance index that should be significantly correlated with the opinion of expert evaluators [6], [9], which is also the underlying objective of our research.

This work targets the detection of the most common deviations to correct reading in the reading aloud task of primary school children (6-10 years old): mispronunciations, false starts, repetitions and intra-word pauses. We approach the task in two steps. In the first step, there is no concern about mispronunciation and only word-relevant segments are retrieved, while allowing repetitions and syllable-based false starts to occur. Subsequently, candidate word segments are classified as incorrectly or correctly pronounced through the combination of several features derived from a phonetic recognition and the likelihood of reference words. This results in an automatic annotation of reading tasks of sentences and pseudowords that can be parsed to evaluate a child's reading performance.

In Section 2, the dataset of children reading aloud will be presented. In Section 3, the first step of segmenting utterances into word-relevant segments while detecting repetitions and false starts is described. Finally, Section 4 details the task of classifying candidate word segments as mispronounced or not, using multiple features and classifiers.

http://dx.doi.org/10.21437/Interspeech.2017-1522

2. Dataset and reading disfluencies

A subset of the LetsRead corpus of European Portuguese children reading aloud [10] is used in this work. Utterances of sentences and pseudowords totalling 10.5 hours are considered. The training set used to train acoustic models, phonetic recognizer and classifiers has 9 hours and a set of 1.5 hours is used to test the system. The children are at primary school level (6-10 years old) and are approximately equally distributed over the grade levels.

The fully transcribed dataset presents varied annotated events for pauses, disfluencies and mispronunciations. Incorrect words are distinguished in two levels: severe mispronunciations or substitutions (SUB) and slight mispronunciations with usually a change in one phoneme only (PHO). From events that represent extra additions beyond word pronunciations, repetitions and false-starts represent 92% of them and are the ones targeted by the method described in Section 3.

3. Segmentation and detection of extra events

As a first step in disfluency detection, repetitions and false starts are targeted while getting a word-level segmentation. With the reference prompt as a starting point, there is no distinction if a word is mispronounced or not, expecting that a word will still be well aligned with its attempt. The word candidate segments can then be further analyzed for pronunciation accuracy in a subsequent step. Standard hidden Markov models were trained with the Kaldi toolkit [11] for this particular stage. The steps taken to get a segmentation for an utterance are:

1. Voice activity detection. Intra-word pauses, usually where a child pronounces a word syllable by syllable, are the most problematic when trying to force-align a word to its pronunciation attempt. Instead of allowing silence after each phone or syllable, we decided to cut all significant non-speech segments (longer than 150 ms) from an utterance to minimize the impact of these cases, even if it leads to unnatural signal transitions. Non-speech segments were found from frame sequences that had a high probability of being silence based on the posterior probabilities output by a phonetic recognizer (described in the following section).
2. Decoding using task-specific lattices. Lattices specifically built from the utterance's original prompt are used to decode the signal, allowing repetitions of words or sequence of words as well as syllable based false starts. The following subsection describes these lattices.
3. Reintroduction of nonspeech segments. After decoding, the segmentation information is reconstructed with the nonspeech segments that were cut in the first step to make the full duration match the original utterance.

3.1. Task lattices

Task-specific finite state transducers (FST) for decoding are built based on the original prompt, either a sentence or an individual pseudoword. For each word of the prompt, a set of elements are added to the lattice: an arc to go back after a word pronunciation allowing repetitions, and an arc that allows multiple false starts. This can be thought of as a forced alignment with some added freedom, or as a constrained decoding. Figure 1 shows an example of the lattice built for the sentence *ele sunhava muito* [ˈelə suɲˈavɐ mˈũȷ̃tu] (he dreamed a lot). False starts are represented by the suffix *PRE*, with multiple pronunciations based on the number of syllables of the word. These pronunciations can be proper prefixes of the word ending at syllable boundaries, which are common interruption points. For the given example: *elePRE* can only be [e]; *sonhavaPRE* can be [su] or [suɲˈa]; *muitoPRE* can only be [mũȷ̃].

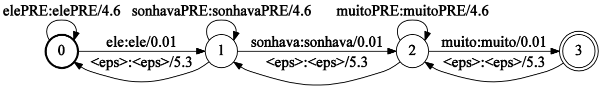

Figure 1: *Schematic of the decoding FST for the prompt "ele sunhava muito".*

The original sentence is obtained by following the horizontal left-to-right arcs. With multiple non-consuming back transitions (<eps>), repetitions of sequences of words such as *ele sonhava ele sonhava muito* are allowed. These are a very common occurrence in the data, representing corrections by restarting at a sentence or clause boundary. The best results were obtained when deletions/skips are not allowed, since there are few in the data where the given prompts are most often realized completely. For a live application, it is conceivable that they should be considered. Insertions are other events that are also not targeted. Using the current methods, a deletion or insertion will often be aligned and then classified as a mispronunciation, so it is still detected that a problem occurs.

3.2. Results

Although the false starts allowed are up to the last syllable, in the transcribed data some are complete mispronunciations of a word. Those are possibly detected as repetitions with these lattices and we decided to analyze the detection of both repetition and false start events as one class. To evaluate the system's performance in detecting these events, we consider that: extra detections (insertions) are false alarms; undetected events (deletions) are misses; events detected as belonging to a different word (substitution) are also misses. These specifications are similar to those used in NIST evaluations [12], though to calculate false alarm rates we divide the number of false alarms by the number of original words. Figure 2 presents the detection error tradeoff (DET) curve obtained by using a wide search beam during decoding and various word insertion penalties and lattice rescoring weights.

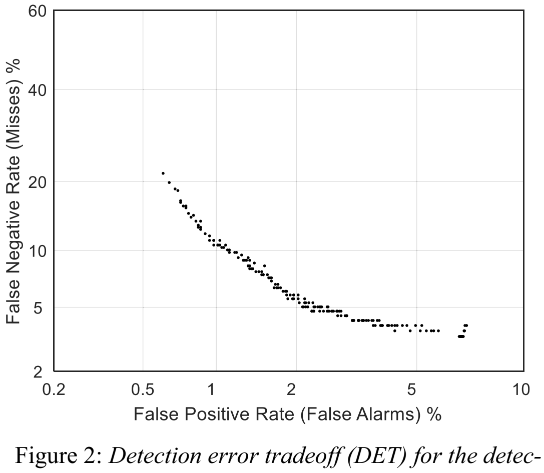

Figure 2: *Detection error tradeoff (DET) for the detection of repetitions and false starts on the test set.*

The word error rate (WER) obtained by using the full text of the original prompts as hypothesis and manual transcription as reference is 9%, with the error corresponding to events of repetitions, false starts, insertions and deletions. Using the weights from the best training result, the WER achieved in the test set is 2.45%, giving an 11.17% miss rate and 0.98% false alarm rate in the detection curve. The optimal point for minimal test WER corresponds to 2.41% WER.

The output of this stage provides time alignment information of candidate word segments, which can then be classified as incorrectly pronounced or not.

4. Mispronunciation classification

We approach the challenge of classifying word pronunciations by defining multiple relevant features and combining them in multi-feature classifiers. A common metric to detect phonetic mispronunciations is goodness of pronunciation (GOP) [13], [14], which computes the likelihood of a phone realization to belong to the ideal phone that should have been pronounced. We compute derivations of GOP-like features on posterior probabilities, edit distances of recognized versus ideal phones and other details about the word.

4.1. Features

For all features that need to consider the reference pronunciation of a word, we allowed multiple acceptable pronunciations as well as co-articulation rules depending on neighboring words (if it was not silence). A neural network based on long-temporal context [15] was trained, outputting posterior probabilities of phones and nonspeech (73% phone error rate with a free-phone-loop model). To recognize the sequence of pronounced phones, we apply a bigram language model derived from the training set. The considered features for a word candidate include:

- A GOP-like accumulated log likelihood ratio (LLR) from a word spotting approach. A candidate word segment may not have the ideal boundary information, either due to segmentation errors or manual transcription flexibility (e.g., including some silence inside the marked boundaries). We previously found success in using a word-spotting approach in the near vicinity of the alignment that finds the peak LLR between the models of ideal word and free phone loop [16], where the most likely boundaries are also discovered.
- Minimum and average GOP (min-GOP and mean-GOP). For a forced alignment of the sequence of phones of the ideal realization of a word, we consider the worst (minimum) likelihood of the aligned phones as a feature as well as the average likelihoods for all phones.
- Maximum and accumulated probability of mismatched phones (maxBadPhnProb, accBadPhnProb). For each recognized phone that does not match the ideal phonetic sequence, we take the average posterior probability over its alignment, and take the maximum and the sum of those values. Hopefully, a mismatched phone with high probability from the recognizer means an increased confidence that a word was mispronounced.
- 3 types of Levenshtein edit distances between recognized and ideal phone sequences: full edit distance (Lev1); edit distance with lower weights for substitutions among phonetic groups (Lev2); edit distance with substitution weights for the phone confusion of the phonetic recognizer (Lev3).
- Difficulty of the word based on dubious and harder pronunciation rules [10] with and without considering word length (Diff1 and Diff2) and OLD20 – the mean Levenshtein distance from a word to its 20 closest orthographic neighbors [17].
- Number of frames of the segment (Nframes), number of phones of the closest allowable pronunciation (Nphones) and number of graphemes (Ngraph).

We also included additional features by exploring normalizations and interactions of LLR with the other features by division or multiplication, represented, e.g., *LLR/Nframes* or *LLR*Lev*.

4.2. Classification Models

Since our defined target indicates whether a word is mispronounced or not, we consider the task a problem of binary classification. If only one feature is analyzed, we can simply define a threshold for a hard decision (*yes* or *no*) or analyze the performance of selecting different thresholds. To combine the information of several features, we explore approaches that either transform the features to a new linear output or make a binary decision:

- Linear discriminant (Linear), by optimizing a linear regression of the features while minimizing the sum of squared errors (SSE).
- Neural networks (NN) with one hidden layer (variable optimum number of neurons) and one linear output trained with scaled conjugate gradient backpropagation and optimizing cross-entropy.
- Support vector machines (SVM) with 2nd order polynomial kernel and C parameter of 0.1.

All models were built and analyzed on the training set using 5-fold cross-validation (CV-train). For predictions on the test set (Test), a model trained over the entire train set is used. For models that depend on random initialization (NN weights and SVM automatic heuristic kernel scale), the best performing one over 10 runs on the training data was selected. Hyperparameters were empirically chosen. To avoid over-fitting to the training set, we also employ stepwise feature selection [18]. Since some features may not provide significant improvement, a feature is selected if the loss in SSE from its inclusion in a linear regression model is statistically significant (in this case, a p-value of an F-statistic test lower than 0.05).

4.3. Results

Two sources of segment boundary information before feature extraction will be analyzed: manual annotation and the automatic annotation described in the previous section. Furthermore, two other analyses are considered: having only severe mispronunciations as the mispronounced/positive class (SUB) or having severe mispronunciations and slight mispronunciations as the positive class (SUB+PHO).

To compare the performance of different classifiers, we must first consider these two aspects: the number of positive samples of mispronunciations is much lower than the number of negative samples for correct words; a false alarm is a more severe occurrence than a miss when evaluating children reading. We found that an F2-score could be a suitable metric for this analysis, similar to an F1-score (harmonic mean of precision and recall) but with higher weight for misses [19], often giving its maximum around 5% false-alarm for our evaluation,

$$F_2 = 5 \cdot \frac{precision \cdot recall}{4 \cdot precision + recall} = \frac{5 \cdot TP}{5 \cdot TP + 4 \cdot FN + FP} \quad (1)$$

where TP are true positives, FN are false negatives and FP are false positives.

Table 2 presents F2-scores for the classification of the SUB+PHO class of some of the best individual features and combination models, with LLR being the best individual feature. There are only slight differences in F2-score among the multi-feature classifiers and all of them gained from feature combination compared with only the best feature.

Table 1: *F2-scores for the classification of SUB+PHO class vs. correct words.*

	CV-train		**Test**	
Classification Model	**Manual**	**Auto**	**Manual**	**Auto**
LLR	**0.687**	**0.675**	**0.645**	**0.639**
min-GOP	0.523	0.524	0.498	0.502
Lev3	0.495	0.504	0.489	0.480
LLR/Nphones	0.651	0.634	0.610	0.605
LLR*Lev3	0.641	0.635	0.611	0.597
LLR*OLD20	0.677	0.666	0.643	0.638
Linear-all	0.710	0.692	0.668	0.652
Linear-stepwise	**0.714**	**0.695**	0.670	0.656
NN-all	0.707	0.692	0.669	0.656
NN-stepwise	0.711	0.690	0.669	0.654
SVM-all	0.700	0.683	**0.676**	**0.658**
SVM-stepwise	0.709	0.694	0.672	**0.658**

From the features selected from stepwise selection, three were consistently chosen (e.g., for all folds of CV-train with automatic labels): LLR, LLR*Lev3, min-GOP. Other features that appear often (for more than 1 fold) are: OLD20, mean-GOP, Nphones. The multi-feature model chosen for further analysis is the linear combination after stepwise selection (Linear-step). Table 3 summarizes results for a 5% false alarm rate, where it can be seen that using manual transcription was slightly better than automatic segmentation and that the combination of features was even more helpful for the SUB class. With automatic segmentation, an improvement from 23% to 21% miss rate is achieved with multiple features for the SUB class, with similar gains for SUB+PHO.

Table 2: *Miss rates for a 5% false alarm rate*

		SUB		**SUB+PHO**	
Labels	**Model**	**CV-train**	**Test**	**CV-train**	**Test**
Manual	LLR	20.78	20.78	27.37	33.51
	Linear-step.	17.02	16.88	23.60	34.03
Auto	LLR	23.12	23.38	28.89	35.84
	Linear-step.	20.86	21.43	26.28	34.81

Figure 3 shows the DET curves for the best individual feature (LLR) and for the linear combination after feature selection, for the cross-validation over the training set. Feature combination appears to be more helpful for 5-10% false alarm rates.

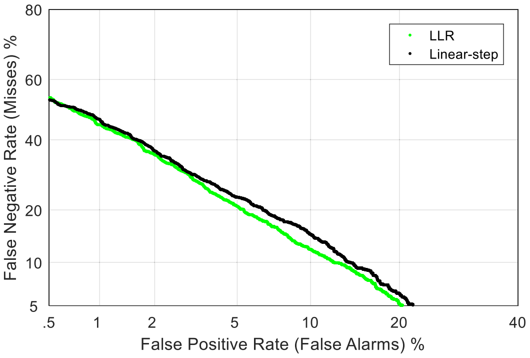

Figure 3: *Detection error tradeoff curve for the classification of SUB class vs. correct words using the best feature (LLR) and a multi-feature model (Linear-step).*

The LLR metric, obtained through a word-spotting approach, although being the best performing feature, can miss mispronunciations where the child added something at the start or end of a word (e.g., plural). Other than that, there are two main issues to tackle to improve results. The first is that the output of the phonetic recognizer is prone to errors, otherwise the match of the recognized phones to reference pronunciation would suffice. The second issue is the subjective manual annotation of correct words and mispronunciations, where many cases are dubious and some special occurrences are very challenging for an automatic system. Most of the false alarm cases for which a very high certainty of mispronunciation was assigned and the manual annotator could hear that the word was in fact correctly pronounced, can be connected to these factors: whispering/non-vocalization where silence is recognized, often occurring in the last word of the sentence or before inhalations; and noise events simultaneous with a word that also lead to either nonspeech or different phones to be recognized.

5. Conclusions

A two-step system was implemented to automatically detect common mispronunciations and disfluencies in children reading. While a low word error rate was obtained with automatic segmentation, the performance of mispronunciation classification suffers slightly when compared to using manual annotation, though the difference is smaller for lower false alarm rates. The combination of features with varying information lead to improved classification results, compared to using only one log likelihood ratio metric.

We attempted to diminish the issue of phonetic recognition accuracy by considering phonetic confusion of the recognizer to calculate edit distance, but this issue could not be solved completely. For future work, to deal with varying reading speeds, the phone insertion penalty for phonetic recognition could be adjusted case by case. We also wish to explore feature interactions or transformations more fully.

6. Acknowledgments

This work was supported in part by Fundação para a Ciência e Tecnologia under the project UID/EEA/50008/2013 (pluri-annual funding in the scope of the LETSREAD project at Instituto de Telecomunicações). The authors acknowledge the support given by Microsoft to this project. Jorge Proença is supported by the SFRH/BD/97204/2013 FCT Grant.

7. References

[1] L. S. Fuchs, D. Fuchs, M. K. Hosp, and J. R. Jenkins, “Oral Reading Fluency as an Indicator of Reading Competence: A Theoretical, Empirical, and Historical Analysis,” *Scientific Studies of Reading*, vol. 5, no. 3, pp. 239–56, 2001.

[2] Y. Liu, E. Shriberg, A. Stolcke, and M. P. Harper, “Comparing HMM, maximum entropy, and conditional random fields for disfluency detection,” in *Proc. Interspeech 2005*, Lisbon, Portugal, 2005, pp. 3313–3316.

[3] H. Medeiros, H. Moniz, F. Batista, I. Trancoso, L. Nunes, and et al., “Disfluency detection based on prosodic features for university lectures.,” in *Proc. Interspeech 2013*, Lyon, France, 2013, pp. 2629–2633.

[4] H. Moniz, F. Batista, A. I. Mata, and I. Trancoso, “Speaking style effects in the production of disfluencies,” *Speech Communication*, vol. 65, pp. 20–35, Nov. 2014.

[5] M. Black, J. Tepperman, S. Lee, P. Price, and S. Narayanan, “Automatic detection and classification of disfluent reading miscues in young children’s speech for the purpose of assessment,” in *Proc. Interspeech 2007*, Antwerp, Belgium, 2007, pp. 206–209.

[6] J. Duchateau, L. Cleuren, H. V. hamme, and P. Ghesquière, “Automatic assessment of children’s reading level,” in *Proc. Interspeech 2007*, Antwerp, Belgium, 2007, pp. 1210–1213.

[7] E. Yilmaz, J. Pelemans, and H. V. hamme, “Automatic assessment of children’s reading with the FLaVoR decoding using a phone confusion model,” in *Proc. Interspeech 2014*, Singapore, 2014, pp. 969–972.

[8] X. Li, Y.-C. Ju, L. Deng, and A. Acero, “Efficient and Robust Language Modeling in an Automatic Children’s Reading Tutor System,” in *Proc. IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2007, vol. 4, pp. 193–196.

[9] M. P. Black, J. Tepperman, and S. S. Narayanan, “Automatic Prediction of Children’s Reading Ability for High-Level Literacy Assessment,” *Trans. Audio, Speech and Lang. Proc.*, vol. 19, no. 4, pp. 1015–1028, May 2011.

[10] J. Proenca, D. Celorico, S. Candeias, C. Lopes, and F. Perdigão, “The LetsRead Corpus of Portuguese Children Reading Aloud for Performance Evaluation,” in *Proc of the 10th edition of the Language Resources and Evaluation Conference (LREC 2016)*, Portorož, Slovenia, 2016.

[11] D. Povey *et al.*, “The Kaldi Speech Recognition Toolkit,” in *IEEE 2011 Workshop on Automatic Speech Recognition and Understanding*, Hilton Waikoloa Village, Big Island, Hawaii, US, 2011.

[12] J. G. Fiscus, J. Ajot, J. S. Garofolo, and G. Doddingtion, “Results of the 2006 spoken term detection evaluation,” in *Proc. SIGIR 2007*, Amsterdam, Netherlands, 2007, vol. 7, pp. 51–57.

[13] S. M. Witt and S. J. Young, “Phone-level pronunciation scoring and assessment for interactive language learning,” *Speech Communication*, vol. 30, no. 2–3, pp. 95–108, Feb. 2000.

[14] T. Pellegrini, L. Fontan, J. Mauclair, J. Farinas, and M. Robert, “The Goodness of Pronunciation algorithm applied to disordered speech,” presented at the The 15th Annual Conference of the International Speech Communication Association - INTERSPEECH 2014, 2014, pp. 1463–1467.

[15] FIT, “Phoneme recognizer based on long temporal context, Brno University of Technology,” 06-May-2015. [Online]. Available: http://speech.fit.vutbr.cz/software/phoneme-recognizer-based-long-temporal-context. [Accessed: 06-May-2015].

[16] A. Veiga, C. Lopes, L. Sá, and F. Perdigão, “Acoustic Similarity Scores for Keyword Spotting,” in *Computational Processing of the Portuguese Language*, J. Baptista, N. Mamede, S. Candeias, I. Paraboni, T. A. S. Pardo, and M. das G. V. Nunes, Eds. Springer International Publishing, 2014, pp. 48–58.

[17] T. Yarkoni, D. Balota, and M. Yap, “Moving beyond Coltheart’s N: A new measure of orthographic similarity,” *Psychonomic Bulletin & Review*, vol. 15, no. 5, pp. 971–979, Oct. 2008.

[18] N. R. Draper and H. Smith, *Applied regression analysis*, 3rd ed. Wiley, 1998.

[19] D. M. Powers, “Evaluation: from precision, recall and F-measure to ROC, informedness, markedness and correlation,” 2011.