Automatic Analysis of Pronunciations for Children with Speech Sound Disorders

Shiran Dudy, Steven Bedrick, Meysam Asgari, and Alexander Kain

Center for Spoken Language Understanding

Oregon Health & Science University

[a] *3181 SW Sam Jackson Park Road Portland*

Abstract

Computer-Assisted Pronunciation Training (CAPT) systems aim to help a child learn the correct pronunciations of words. However, while there are many online commercial CAPT apps, there is no consensus among Speech Language Therapists (SLPs) or non-professionals about which CAPT systems, if any, work well. The prevailing assumption is that practicing with such programs is less reliable and thus does not provide the feedback necessary to allow children to improve their performance. The most common method for assessing pronunciation performance is the Goodness of Pronunciation (GOP) technique. Our paper proposes two new GOP techniques. We have found that pronunciation models that use explicit knowledge about error pronunciation patterns can lead to more accurate classification whether a phoneme was correctly pronounced or not. We evaluate the proposed pronunciation assessment methods against a baseline state of the art GOP approach, and show that the proposed techniques lead to classification performance that is more similar to that of a human expert.

Keywords: Speech Recognition, Goodness of Pronunciation, educational software, diagnostic tools, speech disorders, Support Vector Machine.

URL: dudy@ohsu.edu, bedricks@ohsu.edu, asgari@ohsu.edu, kain@ohsu.edu (Shiran Dudy, Steven Bedrick, Meysam Asgari, and Alexander Kain)

1. Introduction

Phonological disorders are among the most prevalent communicative disabilities diagnosed in preschool and school-age children, accounting for 10% of this population [1]. The American Speech-Language Hearing Association determined that there is an observed relationship between early phonological disorders and subsequent reading, writing, spelling, and mathematical abilities [2]. Furthermore, speech production difficulties affect not only children's communication and academic performance, but also their level of interaction with peers and adults. Considering the limited availability of speech language pathologist (SLPs) [3], it is likely that a parent whose child was diagnosed with a phonological disorder would prefer to have their child practice and acquire language skills as quickly as possible, rather than relying solely on the limited time they can spend with a SLP.

Technology has provided one plausible solution to address the need of improving language skills. Advances in speech recognition technology in the early 90's [4] have attempted to address the problem of the lack of professional human resources to train children, specifically by developing computer assisted pronunciation (CAPT) systems [5] that address children's pronunciation disabilities. These pronunciation systems serve both as assistive tools for diagnosis as well as for practicing correct pronunciations [5, 6, 7, 8, 9, 10, 11].

Unfortunately, many of the current technological tools are still limited. Despite progress in the field of speech processing, existing solutions have primarily concentrated on applying conventional automatic speech recognition (ASR) approaches to assess pronunciation in speech [5, 6, 12]. However, conventional ASR approaches face problems in this area, as the speech of children is characterized by increased acoustic variability, while conventional ASR systems are trained to generate acoustic models from adult speech [13]. Moreover, some mispronunciation patterns may occur in children more frequently than in adults [14]. As a result, while these automatic pronunciation systems hold promise, they are still not widely used since these technologies appear to be less reliable in terms of

their performance [15].

Other solutions have focused on explicitly modeling possible mispronunciations [16, 17, 18, 19]; however, the state-of-the-art system using this approach has the limitation that every target pronunciation must be learned separately against its specific "competing" pronunciations [16] (i. e. mispronunciations). Therefore, the process of developing such a system requires substantial human supervision and input. In addition, the authors mention that in comparison to previous approaches, when introduced with low frequency pronunciation events, the system underperforms, as it requires learning of the specific features of every mispronunciation. The lack of a generalizable approach is a hurdle for automation, as is the need for a large corpus of training examples. An ideal system would incorporate knowledge of mispronunciations from a relatively small corpus while simultaneously require minimal human supervision and input.

Our research goals are derived from these limitations. The short-term objective of our research is to develop a method that will constitute the core component of an effective pronunciation analysis system for children aged 4-12 with speech sound disorders, enabling them to receive accurate feedback on their speech production, even in the absence of a clinician. The desired feedback addresses the question of whether children correctly pronounced a phoneme or not. In addition, to be effective, our system is designed to be highly automated. The long-term goal is to have such a system integrated into remediation techniques, complementing current therapy strategies. In this work, we build upon existing methodologies and extend them. Our main contributions are (1) developing an explicit model of a-priori pronunciation errors for children in the target age range, and (2) explicit modeling of the acoustics of distorted phonemes. We begin by investigating previous approaches in the field of automated pronunciation assessments. Next, we introduce a database containing mispronunciations of children with speech sound disorders. We then describe our proposed approach, and then apply a variety of different evaluation metrics to understand the strengths and weaknesses of our proposed methods. Finally, we present discussion, conclusions, and future work that is planned to further improve the

current research.

2. Background

2.1. Review of Goodness of Pronunciation (GOP)

In this section, we explore in detail several different algorithmic approaches to computing the Goodness of Pronunciation (GOP) technique, followed by an examination of machine learning-based approaches that both contributed to the field and are pertinent to the proposed methods described in Section 4.1.

The GOP technique, originally defined by Witt and Young [5], has been used for pronunciation assessment and has evolved throughout the years in order to improve the quality of the decision algorithm. The basic principle of the GOP technique is to measure the ratio between the likelihood of an *expected* phoneme sequence to the most likely *observed* phoneme sequence. The GOP's outcome is a phoneme level analysis of a pronounced utterance that provides a score for each phoneme representing whether a phoneme was correctly or incorrectly pronounced with respect to the expected word. The first step in the GOP technique is a forced alignment step that segments a recording using an automatic speech recognition (ASR) system. This step forces the ASR to recognize a predetermined phonemic sequence of the *expected* word. The result is the time locations (marking a phoneme segment boundaries) corresponding to the utterance start-time of each phoneme in the sequence. The phoneme segmentation step is essential as every phoneme is assessed separately. The second step is demonstrated in Equation 1:

$$GOP(q_i) = \log(P(q_i|\mathbf{O}))/N_{q_i} = \log\left(\frac{P(\mathbf{O}|q_i)P(q_i)}{\sum_{j\epsilon J} P(\mathbf{O}|q_j)P(q_j)}\right)/N_{q_i} \qquad (1)$$

where

$$q_i \quad - \quad \text{the expected phoneme} \tag{2}$$

$$\mathbf{O} \quad - \quad \text{the observation (phoneme segment)} \tag{3}$$

$$\{q_j\}_1^J \quad - \quad \text{phoneme set} \tag{4}$$

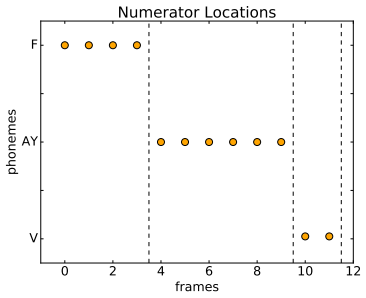
$$N_{q_i} \quad - \quad \text{duration of observed phoneme} \tag{5}$$

are the equation's components. The likelihoods of a segment's acoustic features are extracted with respect to the expected phoneme in the numerator of Equation 1. A similar process takes place in the denominator only that instead of a single likelihood, the likelihoods of the observation (phoneme segment) are extracted with respect to the acoustic models of every phoneme in the set and are summed. In addition, phoneme durations are normalized in the GOP in order to have a robust system that accommodates different phoneme lengths which would account for phoneme duration variation in different participants.

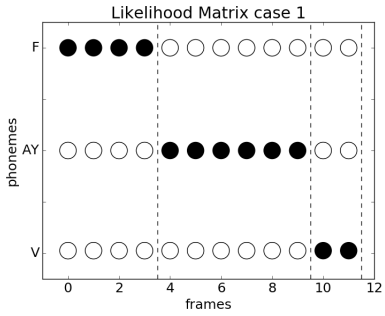
There are two possible outcomes of the process as described. If the denominator is dominated by the likelihoods of the same phoneme model as in the numerator then, the GOP's numerator-to-denominator ratio may exceed a predetermined threshold leading to a GOP decision of "correctly pronounced phoneme". Conversely, a GOP score below the threshold determines an "incorrect pronunciation".

The outcome of applying different GOP techniques is examined by illustrating four case studies of typical data. Figure 1 introduces case studies in which a child was required to utter the word 'five'; one scenario in which she correctly pronounced it and a one in which she did not. The numerator segments in Figure 1a are the result of the phoneme segmentation step for the expected word 'five' and correspond to each phoneme (in order) of the word 'five' (see y-axis). Case Study 1 and 2, in Figures 1b and, 1c, describe high likelihoods for the words 'five' and 'vive' respectively (marked in black circles).

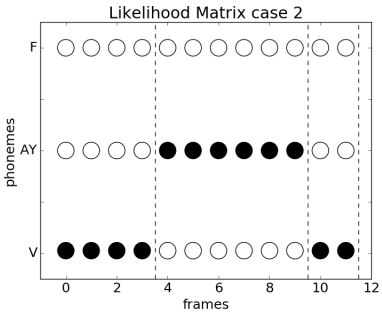
The second step of applying the GOP method is shown in Figure 2. A numerator-denominator ratio is calculated by dividing the likelihood's sum of



(a) Numerator



(b) Case 1

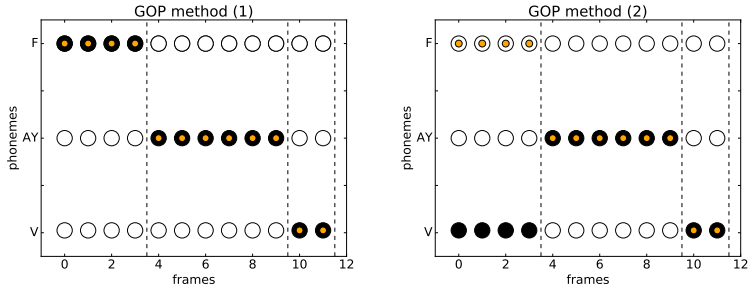


(c) Case 2

Figure 1: Case studies representation. 1a refers to phoneme locations of segmentation step. 1b and 1c describe high phoneme likelihoods in black.

the expected phoneme 'F' (in the first phoneme segment for instance) to the sum of likelihoods of all phonemes found in the set found in the denominator. In the first phoneme segment of Case Study 1, the denominator is dominated by the sum of likelihoods of 'F' (as they are marked in black circles) and since the numerator is the sum of likelihoods of 'F', the ratio will probably exceed a predetermined threshold resulting in a 'correctly pronounced' phoneme decision (Figure 2a). Conversely, Case Study 2's denominator is dominated by 'V's likelihoods, which probably will lead to a ratio below the threshold resulting in an 'incorrectly pronounced' phoneme decision (Figure 2b).

A second version of GOP introduced by Witt and Young [20] is a simplification of the original GOP – GOP Max. The motivation was to provide a more accurate estimation of the GOP ratio by modifying the denominator computa-



(a) Case 1 (b) Case 2

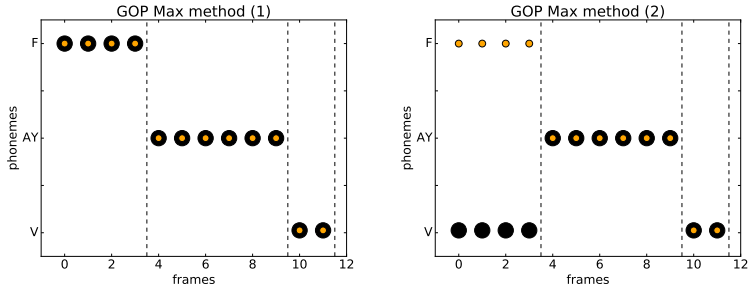
Figure 2: Case studies with GOP method applied

tion. Since the denominator addresses the question of what is the most likely phoneme, it is unnecessary to sum of all the phonemes' likelihoods of the target set (within a segment). Instead, in GOP Max the denominator is the likelihood only of the most likely phoneme – the phoneme with the highest (maximum) likelihoods that stretches over the longest duration (within a segment). This is described in Equation 6.

$$GOP_{max}(q_i) = \log \left(\frac{P(\mathbf{O}|q_i)P(q_i)}{\max\limits_{j \epsilon J} P(\mathbf{O}|q_j)P(q_j)} \right) / N_{q_i} \tag{6}$$

Equation 6 differs from Equation 1 in the denominator. In Equation 1, the denominator is the sum of all phoneme likelihoods, where as Equation 6 simply uses the single most likely phoneme. GOP Max was further explored in Witt and Young [5] and Mak et al. [21].

GOP Max is applied in Figure 3. In the first segment of Case Study 1, the denominator is the summation of the likelihoods of 'F' which is equal to the numerator's value (likelihoods of 'F' as well) resulting in a numerator-denominator ratio of 1. Thus, the GOP's decision, given this ratio, is of a 'correctly pronounced' phoneme. On the other hand, the divider of the numerator in Case Study 2 is the sum of likelihoods of 'V'. Since the numerator's sum is small ('F' not circled in black), the ratio is expected to be less than 1 resulting in a decision of an 'incorrectly pronounced' phoneme.

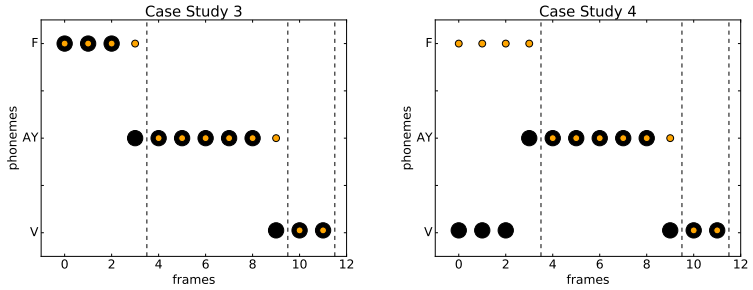


(a) Case 1 (b) Case 2

Figure 3: Case studies with GOP Max applied

While GOP Max provides a more accurate version of pronunciation assessment than of the original GOP, it has limitations. The assumption that every segment is represented by a single phoneme is not always aligned with the ASR recognition output, as there can be more than one phoneme uttered by the speaker. Determining that the phoneme with the longest maximum-likelihood duration is the phoneme in the denominator, as presented in GOP Max, could be problematic particularly when this phoneme's duration is slightly greater than 50% of the segment duration. Representing an entire segment by a phoneme that only half of the time (or slightly more) produced the highest likelihoods might not reflect accurately the actual observation and is likely to lead to an erroneous pronunciation assessment. In order to illustrate this limitation Figure 4 provides Case Study 3 and 4 for the words 'five' and 'vive' respectively. Figure 4 shows that the highest likelihood's paths throughout the segments are not necessarily aligned with the segmentation borders determined by the ASR. In other words, the black circles do not match the boundaries imposed by the vertical lines. In both 4a and 4b, a different phoneme appears (AY's black circle) on the last frame of the first segment.

To re-examine the GOP Max assumption of having only a single phoneme represented in a segment (observation), a third version of the GOP was introduced by Song et al. [22] called LGOP (Lattice based GOP). Song et al. [22]



(a) Case 3 (b) Case 4

Figure 4: Case studies representation of relaxed conditions

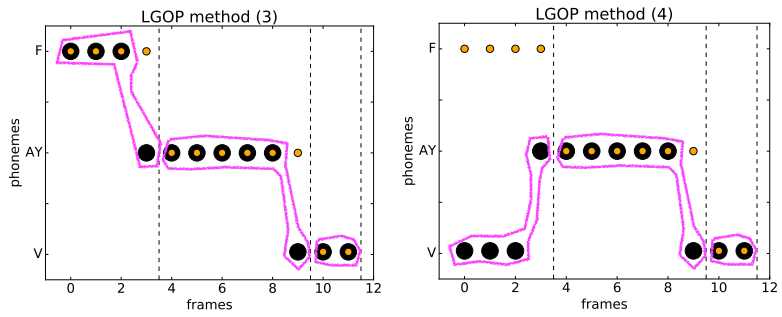
relaxed this assumption arguing that ASR likelihood's output does not always match constrained forced alignment process (GOP's first step). In other words, the ASR can assign high likelihoods to other phonemes within the segment which results in more authentic representation of the participants' utterances. In LGOP (Lattice based GOP) [22] likelihoods in the denominator are based on the highest likelihood's path within a segment permitting more than one phoneme to compose the sum. (Although one phoneme per frame is allowed). The denominator's final summation is dividing the numerator as shown in Equation 7. The denominator sums the phoneme likelihoods determined by a Viterbi process [23] shown in Equation 8.

$$LGOP(q_i) = \log\left(\frac{P(\mathbf{O}|q_i)P(q_i)}{\sum_{t\epsilon T} P(q_j|V_t)}\right)/N_{q_i} \tag{7}$$

$$V_t = \max_{i,j\epsilon J}(V_{t-1}\alpha_{ij}q_i(o_t)) \tag{8}$$

In Figure 5 LGOP is applied to Case Study 3 and 4. The likelihoods composing the denominator's sum are circled. The numerator does not have the exact value as the denominator since its last frame is not of 'F'. While this method can effectively present this small mismatch in the last frame (resulting in a ratio slightly smaller than 1), the LGOP decision will still be of a "correct pronunci-

ation" (because it is dominated by the same phoneme in the denominator and numerator). Had the likelihoods been as presented in Figure 1b, the numerator-denominator ratio would have been 1, similar to GOP Max. The decision in Case Study 4 will be, as expected, an "incorrect pronunciation" (because the denominator is not dominated by 'F').



(a) Case 3 (b) Case 4

Figure 5: Case studies with Lattice-Based GOP applied

Since LGOP provides a more authentic segment representation, its outcome is more accurate. LGOP generalizes on GOP Max represented in Equation 6, since it captures both when the best phoneme path was made by 'walking' on the same phoneme model (Case Study 1 and 2) and when the best path was determined by more than one phoneme model (Case Study 3 and 4).

This review of the GOP approach lays the foundation for the current work. While LGOP is a more accurate solution than its earlier versions, it has limitations as well. One limitation is its acoustic features. Often, the phoneme set is composed of acoustic models of correctly pronounced phonemes, that imposes limited mapping of an observation to a phoneme. In other words, a phoneme set containing acoustic models of non-canonical realizations of phonemes as well might assess more accurately phoneme quality. Another possible drawback that can limit the performance is not incorporating knowledge on repetitive/common mispronunciation mistakes that can potentially provide improved assessments. (These statements are explained and supported in the next paragraph)

One extension of the GOP that addresses mispronunciations was introduced by Doremalen et al. [16], who conducted a research focusing on the most confusable vowels in the Dutch language by a second language learners (L2). The Weighted Phone Confidence (wPC) approach was presented in Doremalen et al. [16], whose paper is based on Phone Confidence (PC) scores. Equation 9 describes the PC score that is generated for each p_i. The set of p_is contains the confusable and target pronunciations for a particular target phoneme.

$$\mathrm{PC}_{p_i}^{p_{targ}} = \frac{1}{t_e - t_b} \sum_{t=t_b}^{t_e} \log \left(\frac{P(O_t|p_{targ})}{P(O_t|p_{targ}) + P(O_t|p_i)} \right) \tag{9}$$

In Equation 9, the likelihood of the target phoneme is found in the numerator and denominator of the PC component, and in the denominator, this likelihood is added to the likelihood of the possible realization p_i. For every target phoneme the PC scores are assigned to a logistic regression model with specific weights (βs in Equation 10) associated with each p_i's PC score (pronunciation realization) as shown in Equation 10.

$$\mathrm{wPC}^{P_{targ}} = \frac{1}{1 + \exp\{-(\beta_0 + \sum_i \beta_i PC_{p_i}^{p_{targ}})\}} \tag{10}$$

The wPC score provides a decision to whether a phoneme was correctly pronounced or not. Doremalen et al. [16]'s research incorporates knowledge on pronunciation errors by taking into account the different characteristics of every phoneme. While results were promising, one limitation this approach has is the thorough investigation required to extract the specific list of "competing" phonemes for every target phoneme found in the set. A second limitation, involves the learning of the particular weights of every PC-realization. Since p_is are different for every target phoneme, a different set of weights is learned separately for every target phoneme for the wPC decision algorithm. On the one hand, wPC incorporates knowledge of mispronunciation errors and carefully handles every phoneme in the target set, but on the other, it requires large sets of annotated data for every language of origin of an L2 population to accurately derive meaningful weights from as mentioned in [16], and as such is less "au-

tomatic" or "human-free" as it requires more intervention in the process than GOP, GOP-Max, and LGOP.

One of the goals of the current research is to present a new version of the GOP that aims to improve upon LGOP method by addressing the main limitations of the above-presented approaches. Unlike the LGOP, the proposed systems are able to incorporate knowledge of common pronunciation realizations of a target phoneme. Notably, though, the mechanism to generate both of the proposed systems is automated, in order to have a minimal human interference in the process and in order to generalize its decision process to every target phoneme that is learned (as opposed to wPC approach). Our experiment uses the current state-of-the art version of GOP (LGOP) as a baseline against which to compare the current research's two proposed methods.

2.2. Review of Machine Learning Related Research

Another family of approaches to pronounciation assessment relies on machine learning techniques. One system describes a particular concept of incorporating mispronunciation phonemes into the decision process. Other systems draw on methods of Decision Trees and Support Vector Machines.

Ronanki et al. [17] proposed to incorporate mispronunciation phonemes into the decision process and address mispronunciation in a more constrained fashion. First, the audio file was force-aligned against an expected word. Then, each phoneme segment was sent for recognition not only by the phoneme model, but also by its potential mispronounced phoneme models. This approach is applied to a word level and a phrase level recognition. An example of a word level recognition, from the paper [17], was for the word "WITH":

<phonelist> = ((W | L | Y) (IH) (TH))

<phonelist> = ((W) (IH | IY | AX | EH) (TH))

<phonelist> = ((W) (IH) (TH | S | DH | F | HH))

In the example, at every segment of the word there are several phoneme candidates the system can choose from when recognizing it with an ASR. While the

process of deriving potential mispronounced phonemes was not clear (as it may have been extracted from TIMIT corpus, set by a linguist, or by another alternative), limiting the possible number of mispronunciations reduces the phoneme search space and the computational complexity. One of the proposed methods of the current research incorporates knowledge of mispronunciation phoneme candidates by using data-driven models to extract mispronunciation patterns. In addition, this proposed method addresses the issue of unseen mispronounced phonemes as well.

Decision Trees [24] have been employed in assessing phonemes pronunciation quality as well. Peabody [25] applied a decision tree that incorporated novel acoustic features to detect vowel mispronunciations. KullbackLeibler and Bhattacharya were applied as distance metrics to represent how close a phoneme was to a non-native pronunciation and to a native one. The motivation for using such a decision tree was mainly the ability to understand the decision process (of the phoneme assessment) and the reasoning behind it. Previously, Zechner et al. [26] focused on multiple regression training to demonstrate the high correlation of the described tree decisions with human annotators. To build the tree, Zechner et al. [26] introduced various features that described fluency of speech such as speaking rate and articulation rate. Minematsu [27] worked on detecting mispronunciation of English by Japanese natives. The described approach employed Bhattacharya distance to syllable units with dynamic programming, which determined how close a spoken word was to a correctly pronounced one. In addition, Minematsu [27] constructed a phonological decision tree, such that when a subject's phonemic sequence was determined, comparing the distance of the mispronounced phoneme from the expected position measured how intelligible the word was.

Support Vector Machine (SVM) [28] has been applied by Wei et al. [18] as another machine learning technique aimed at developing decision systems for pronunciation classification. The classifier's input was a feature vector on a phoneme level described by Log Likelihood Ratio distance - LLR distance in

Equation 11:

$$LLR(\mathbf{O}|q, q_i) = \log P(\mathbf{O}|q) - \log P(\mathbf{O}|q_i) \tag{11}$$

LLR is undertaken separately for each model. The expected phoneme model, given the observation, is compared with a phoneme model in the target set. The feature vector that is fed into the SVM includes the likelihoods of every model in the phoneme set, as shown in Equation 12. The number of classifiers is equal to the number of phonemes.

$$f = [LLR(\mathbf{O}|q, q_1), LLR(\mathbf{O}|q, q_2), ..., LLR(\mathbf{O}|q, q_N)] \tag{12}$$

In the same paper, Wei et al. [18] also defined a Pronunciation Space Model (PSM). This space is built by classifying all observations of the same phoneme in an unsupervised fashion in order to find pronunciation patterns for mispronounced phonemes as well as for correctly pronounced ones. This method is described in Equation 13.

$$f_{PSM} = [LLR(\mathbf{O}|q, q_{1,1}), LLR(\mathbf{O}|q, q_{1,2}), ..., LLR(\mathbf{O}|q, q_{N,K})] \tag{13}$$

Each phoneme model q_N from Equation 12 is replaced with $q_{N,1-K}$, where N represents the phoneme group as in Equation 12 and $1-K$ represents the specific phoneme patterns resulting from the PSM process. Overall, SVM/PSM had a higher performance when compared to the SVM and original GOP method [18].

Another machine-learning approach described by Strik et al. [19] aimed to focus on a specific phone mispronunciation. Strik et al. [19] applied Linear Discriminant Analysis (LDA) to two different feature sets; one with MFCC and the other with Acoustic-Phonetic Features (APF) based on the approach described by Weigelt et al. [29]. Our choice of the Weigelt algorithm was motivated by its ability to discriminate between voiceless fricatives and voiceless plosives; indeed Strik et al. [19]'s focus was on distinguishing between fricative /x/ from plosive /k/. While both phonemes' spectral envelope is similar, their amplitude is different. That was the motivation for using energy's first moment, zero

crossing rates, and relative energy in the peak surroundings as acoustic features. APF-LDA was introduced and compared with an MFCC-LDA. Both had similar performance and showed significantly better performance than the original GOP.

2.3. Research Proposals

There is a tension between algorithm-based and machine learning based approaches in the field of automatic pronunciation assessment. This research describes an algorithmic approach with a newer version of the GOP and a machine learning approach with an SVM classifier to measure the quality of phoneme pronunciations. Both proposals are compared against the latest state-of-the art GOP – LGOP approach. In addition, the features that are used are derived from speech that contained authentic mispronunciations as well as correct pronunciations in order to observe closely how well the expected phoneme was pronounced. It is our hope that this will provide a more naturalistic and ecologically valid foundation to pronounciation analysis.

3. Data

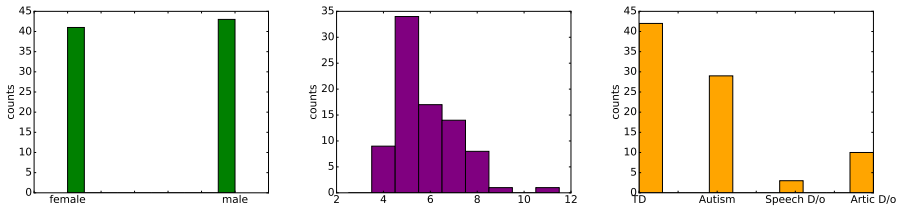
3.1. Data Collection

The data that was predominantly used in the experiment came from the Corpus of Children's Pronunciation (CCP). In order to collect the data, we recruited 86 children aged 4-12 ($\mu = 5.3$, $\sigma = 1.3$). Co-occurrence of receptive and expressive language disorders is prevalent in children with speech production challenges, and so these children were screened to ensure that they had the ability to complete the tasks required in the study. The diagnosis of a speech-sound disorder was conferred by a licensed, credentialed speech-language pathologist who completed a standardized assessment, and/or exercised clinical judgment based upon transcribed speech samples and normative data. Additional requirements, such as receptive/expressive language skills and the behavioral capacity to complete the necessary tasks, were also taken into consideration by the Speech-Language Pathologist (SLP).

Children spoke words from the Goldman-Fristoe Test of Articulation (Sounds-in-Words Section only) [30], consisting of 53 simple words (e.g. “house”, “tree”, “window”). Productions were elicited from images with the assistance of an SLP.

Phonetic segmentation was performed using Praat [31] by a second SLP.

A senior, expert SLP (i.e., a third SLP) phonetically transcribed the children’s speech (with simultaneous access to video) using the full range of International Phonetic Alphabet (IPA), including a wide variety of diacritics to represent distorted symbols. The third expert also scored whether a phoneme was pronounced correctly, or incorrectly. For some words, several canonical pronunciations were acceptable, and thus actual pronunciations were compared to their closest canonical pronunciation. The final outcome produced for each audio file was the transcription of the uttered word, its segmentation to phonemes and a score for each phoneme in the word.



(a) Gender distribution (b) Age distribution (years) (c) Diagnosis distribution

Figure 6: CPP corpus statistics

Three graphical representations of the distribution of the subjects in the Corpus of Children’s Pronunciation are shown in Figure 6. This corpus contains a balanced gender distribution, as seen in Figure 6a. Figure 6b indicates a majority of children in the age range of 5–6 years, and a minority in the range 11–12 years. Figure 6c shows four subjects that have speech disorders, 11 subjects that have an articulation disorder, 28 with an autism spectrum disorder, and 43 who are typically developed.

3.2. Data Analysis

The main goal of this subsection is to analyze the CCP corpus presented in subsection 3.1. In order to produce the best-fitting phoneme sequence corresponding to speech pronunciation, each word expressed in the audio file was annotated as described in 3.1. This enabled us to identify common patterns of phonemes that are referred to as trends.

#	exp.	rec.	%
1	ɹ	w	1.060
2	ʒ	dʒ	0.625
3	ʃ	tʃ	0.605
4	l	w	0.514
5	θ	f	0.393
6	s	θ	0.363

(a) typically-developing group

#	exp.	rec.	%
1	ɹ	w	1.968
2	l	w	1.018
3	ŋ	n	0.514
4	k	t	0.514
5	θ	f	0.484
6	s	θ	0.395

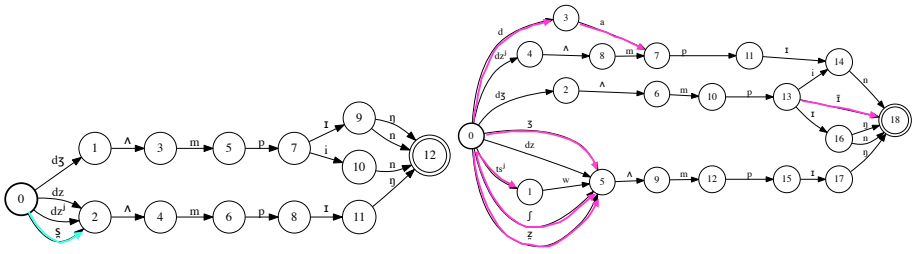
(b) speech-disorder group

Table 1: The top six phoneme-confusions.

In the first part of the analysis, we detected phoneme level trends occurring in both groups: children who are typically-developed (TD) and children who have speech disorders (SD). The top 6 trends were extracted from each group's confusion matrix and presented in Table 1. The table describes the expected phoneme and the phoneme recognized by an expert referred to as exp. and rec., respectively. In addition, the % column in Table 1 presents information associated with the percentage of each confusion, which was computed by the number of cases of the particular confusion divided by the total number of pronounced phonemes. As anticipated, all confused sounds in Table 1 were phonetically close to the expected sounds. Four confusion patterns were shared by both groups (TD and SD), though to a different extent. High confusion rates were observed in the first, fifth, and sixth patterns (the same rank in both groups) but also in pattern 4 in Table 1a and 2 in Table 1b. These confusions

might suggest that their expected phonemes are particularly 'hard to pronounce' phonemes rather than confusion trends that can be identified with TD or SD groups. At the bottom of the table, there is a similar rate range of confusion between the groups, while at the top, the confusion rate is twice as large in the speech disordered group 1b. Smit [32]'s research on speech of typically developed children defined the pattern of /ŋ/ →/n/ (shown in Table 1b) as a "common mismatch". This finding might suggest that /ŋ/ →/n/ is not a unique pattern that characterizes the SD group. Furthermore, a 1960 study conducted by Graham and House [33], discovered a number of patterns that were often confused by children. He performed a listening test in which children were required to determine whether two phonemes sound 'the same' or 'different'. Some of his most commonly confused patterns were: /ɹ/ →/w/, /l/→/w/, /θ/→/f/, /s/→/θ/, and /k/→/t/ (shown in Table 1a and Table 1b).

In the next part of the analysis, we inspected the word level patterns. In Figures 7a and 7b, while the TD group demonstrated a few phoneme substitutions with relative acoustic proximity, SD subjects expressed this phenomenon to a greater extent along with phoneme deletion events. The SD group also had more variety of confusions.



(a) typically-developing group (b) speech-disorder group

Figure 7: Comprehensive pronunciation graphs of the word "jumping". The cyan and magenta arrows mark phonemes patterns unique to the group.

This data analysis provided a greater understanding of the children's confusion trends, and confusion frequencies found in the CCP corpus. The ways in which the various correct and incorrect pronunciations were incorporated into

the proposed methods are described in the next section.

4. Method

4.1. Approaches

4.1.1. GOP-CI

Similar to our previous work [34], we introduce the following algorithmic approach, which aims to improve the baseline GOP measure described in Equation 7 of the LGOP approach by (1) learning acoustic models from a large children's speech database aged 3–15 and then adapting to the speech of children in the final target age range of 4–11, (2) incorporating an explicit model of correct and incorrect pronunciations of the corpus described in the previous section, and (3) explicit modeling of the acoustics of distorted phonemes through the availability of fine-grained phonetic transcriptions during recognizer training.

For a given target word w, composed of P phones $p_1, p_2, \ldots, p_P$, let $b_1, b_2, \ldots, b_P, b_{P+1}$ denote the phoneme boundaries (in frames), such that p_i spans frames $[b_i : b_{i+1})$ (half-open interval). We estimate phoneme boundaries and frame-level likelihoods through ASR lattices created by the Kaldi toolkit [35]. These lattices are created based on Weighted Finite State Transducers (WFSTs) [36], which efficiently integrate the sources of knowledge of the acoustic model, the language model, and the lexicon during the decoding phase of the ASR system. We define the improved GOP measure for the i^{th} phoneme, p_i, of the target word w, as

$$\text{GOP-CI}\ (p_i) = \frac{L\left(\varphi^*_{\text{C}}[b_i : b_{i+1})\right)}{\alpha L\left(\varphi^*_{\text{CI}}[b_i : b_{i+1})\right) + (1-\alpha) L\left(\varphi^*[b_i : b_{i+1})\right)} \tag{14}$$

where

$$\varphi^*_{\text{C}} = \arg_\varphi \max\left(\mathcal{H} \circ \mathcal{C} \circ \mathcal{L}_{\text{C}}\right) \tag{15}$$

$$\varphi^*_{\text{CI}} = \arg_\varphi \max\left(\mathcal{H} \circ \mathcal{C} \circ \mathcal{L}_{\text{CI}}\right) \tag{16}$$

$$\varphi^* = \arg_\varphi \max\left(\mathcal{H} \circ \mathcal{C}\right) \tag{17}$$

represent the most likely Viterbi path of phoneme sequences given different the WFST networks $\mathcal{H}$, $\mathcal{C}$, and $\mathcal{L}$, which denote the phone transition lattice

(Hidden Markov Model-based), triphone transition lattice (context-dependent), and a syllable transition lattice given a vocabulary (lexicon) (discussed further in the next paragraph), respectively. The symbol $\circ$ denotes WFST composition, and $L(\cdot)$ represents the summation of negated log-likelihoods over associated frames. The tuning parameter α controls the contribution of likelihood scores driven from constrained lattice in the denominator and one open-loop lattice. In other words, it controls the degree to which we expect to encounter previously seen pronunciation mistakes.

We employ both constrained and open-loop lattices with identical $\mathcal{H}$ and $\mathcal{C}$ in order to compute the GOP(p_i) with the best path (see Equation 7). The two constrained lattices, located in the numerator and on the left-hand side of the denominator of the GOP, are generated by composing $\mathcal{H} \circ \mathcal{C}$ with either the lexicon containing correct pronunciations for the target words, $\mathcal{L}_{\mathrm{C}}$, in Equation 15, or the combination of correct and incorrect pronunciations, $\mathcal{L}_{\mathrm{CI}}$, in Equation 16. Correct pronunciations were globally constructed from all available data, whereas incorrect pronunciations were sourced from the training set exclusively. Similar to the forced alignment step described in Eq. 1, phone boundaries are identified using Equation 15 dictating which frames' likelihoods are summed up to produce the likelihood score (On the right-hand side of Equation 14, subscript i represents the phoneme boundaries extracted from the numerator by Equation 15). The open-loop lattice is necessary in order to account for the possibility of encountering previously unseen mispronunciations, or even entirely unexpected words. It is created by composing $\mathcal{H} \circ \mathcal{C}$ as shown in Equation 17 without confining it to any vocabulary from $\mathcal{L}$ to enable any triphone sequence combination. The open-loop component is located on the right-hand side of the denominator of Equation 14. The reason for naming the described algorithmic approach as GOP-CI is due to its CI component that stands for Correct and Incorrect patterns observed in the corpus that is integrated in to the formula.

4.1.2. GOP-SVM

GOP-SVM is the second proposed method that was employed following the algorithm-based results. This approach involved an SVM (Support Vector Machine) [28] for learning the different classes for "correct" and "incorrect" decisions. We incorporated the likelihoods extracted from the GOP algorithm, specifically from Equations 15, 16, and 17. These likelihoods composed the input feature vector for the SVM. A weighted SVM was applied to overcome the unbalanced class distribution in training data favoring correct pronunciations over incorrect ones. We trained SVM classifiers with several kernel functions including linear, polynomial, and radial basis function (RBF) employed from open-source Scikit-learn toolkit [37]. Parameters of the optimal SVM model were determined on the development set separately for each fold via grid search. Experimental results showed that SVM with RBF kernel along with a global C variable of 0.001 outperformed other SVM classifiers and thus these were the characteristics of the SVM method we propose.

4.2. Training

Learning acoustic models in ASR systems requires a fairly large amount of training data, which is mostly beyond the scope of data collection for specialized populations. In order to tackle this issue, a large children's speech database was employed in addition to our small corpus for learning acoustic models. We built a context-dependent HMM-GMM (Hidden Markov Models-Gaussian Mixture Models) system based on speech utterances from the OGI Kids Corpus [38] and Corpus of Children's Pronunciations (CCP) introduced in Section 3.1. The OGI Kids Corpus is composed of 27 hours of spontaneous speech from a gender-balanced group of 1100 typically developed children from kindergarten through to grade 10 [38]. For extracting speech features, a window of 7 frames (current frame, 3 prior and 3 previous frames) was taken to extract 13-dimensional MFCCs with delta and delta-delta coefficients. After undertaking cepstral mean and variance normalization for each speaker, features were reduced down to 40 dimension using linear discriminant analysis (LDA). Model-space adaptation,

using maximum likelihood linear regression (MLLR), was applied, followed by speaker adaptive training (SAT) of the acoustic models by both vocal tract length normalization (VTLN) and feature-space adaptation using feature-space MLLR (fMLLR).

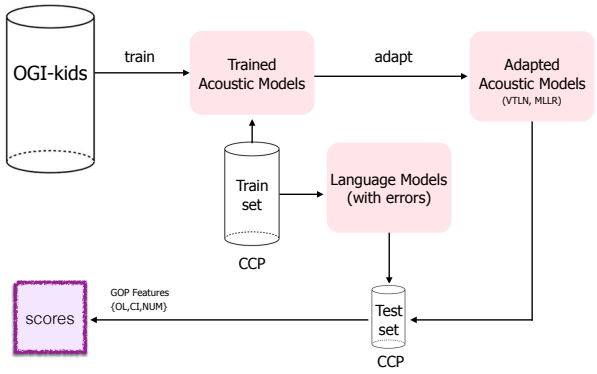


Figure 8: Block diagram for proposed methods

The general structure of the process is shown in Figure 8. The OGI Kids Corpus contains speech from typically developed subjects and its purpose in this research was to build strong acoustic models for children. Both OGI corpus and CCP corpus were trained together to generate the acoustic models of the system. While CCP corpus was smaller than OGI's it contained a greater variety of acoustic models that enriched the learning process and helped in recognizing pronunciations. However, as this corpus was smaller, its impact on establishing good acoustic models that were not present in OGI Kids Corpus was limited. In addition, annotations were extracted for all pronunciations in CCP to generate the language models. Finally, given a test utterance, the information from both annotations and acoustics was incorporated to score each phoneme.

For evaluation purposes, five-fold cross validation scheme was used by dividing the CCP into five subject-independent sets. For training the ASR model parameter, four of the five of CCP sets were used in addition to the OGI Kids Corpus for training. The fifth part was divided into the following: 0.25 was used for development, and 0.75 used for testing and reporting the performance.

4.3. Decoding

For each expected word, we defined a search FST that includes both the (possibly multiple) a-priori good pronunciations and the known bad pronunciations (phoneme sequences), as seen during the training. We decoded the expected word using both the FST and an open-loop (OL) phoneme recognizer.

In the decoding graph, context-dependent and context-independent state identifiers are the input symbols of the lattices in Equations 15, 16 and 17, and the sequence of phonemes are the output symbols. In order to compute the GOP for each expected word, the most likely sequence of phonemes was obtained using the Viterbi algorithm for each of the elements: C (of correct patterns in numerator) , CI (of correct and incorrect patterns), and OL (which is the context independent element). For each expected word there were between 1 and 3 different correct sequences of pronunciation for the word representing the C patterns in the numerator. An example is the word "ball" that can be correctly pronounced /b//a//l/ but also /b//ʌ//l. Next, because CI included more patterns than C, a greater flexibility was allowed to produce the most likely sequence from CI patterns. CI group included only observed sequences (data-driven). The last output is OL (context-independent) pattern recognition that was not subjected to observed sequences. The negated log-likelihoods of each of the three components (C, CI, and OL) were computed separately per boundary and applied to Equation 14.

4.4. Development Set

In our experiment we examined three methods: GOP-CI presented in Eq. 14, the GOP-SVM presented in Section 4.1.2 and LGOP in Eq. 7 as a baseline approach. The development set found in each fold was used for finding the best θ for GOP-CI and LGOP approaches, that determined the threshold from which higher ratios were classified as "correct" pronunciations and lower ones were classified as "incorrect" ones. The choice of θ was made separately for each fold and for each method. The development set was used to find α for GOP-CI method as it enabled choosing the optimal linear combination of the constrained

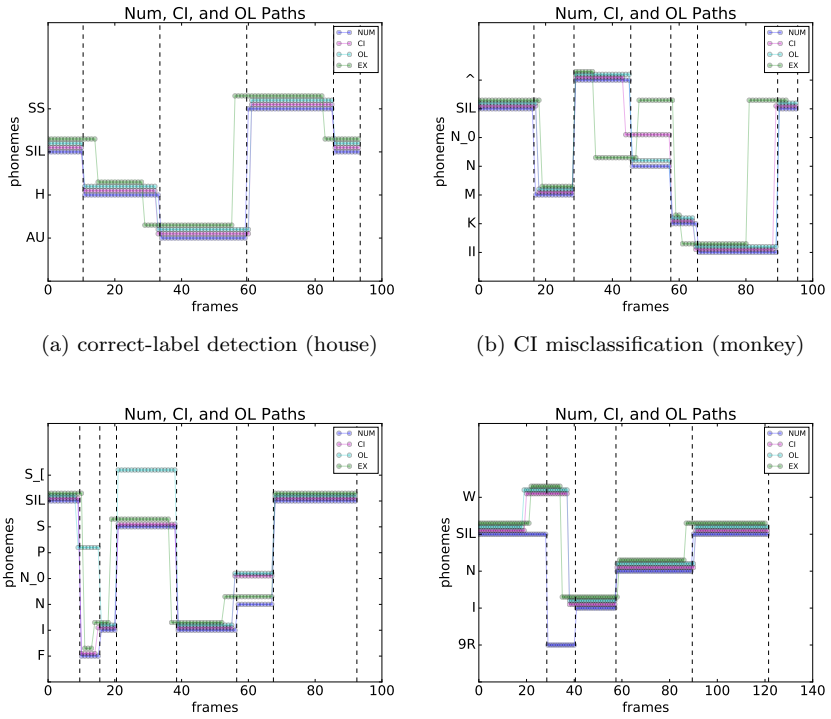
lattice (CI) and open loop (OL) described in Equations 16 and 17 respectively. In addition, by setting α for words that were not seen during training phase (seen only in development set) it was more robust to unseen data (of test set). Finally, for GOP-SVM approach, we extracted an optimal C value for the SVM kernel from the development set.

5. Results

5.1. Raw Results

In the first part of data analysis the raw results produced by the ASR are inspected. To visualize the recognition results by each of the elements of Equations 15, 16, and 17 found in the GOP-CI and GOP-SVM approach, we extracted authentic examples from our data. In Figure 9, there are four cases produced by our system. These cases show the segmentation of phonemes across the different elements; namely, the numerator (with the correct path) - "Num", the correct-incorrect path "CI," the Open Loop path "OL", and the human annotator's decisions "EX". Though they are important for computation, the actual likelihoods of each frame are not shown here because the focus of these figures is on the phoneme decisions and the boundaries that are the result of the highest likelihood pattern for every component of the proposed method described in this paragraph. Understanding these cases is essential for understanding decision processes of GOP-CI and LGOP methods.

In 9a, the utterance was "house." Every component of the system ('NUM', 'CI', 'OL') recognized the phonemes that were recognized by the annotator. The phoneme boundaries were consistent across all components. However, within the boundaries, there is a mismatch among all of the components ('NUM', 'CI', 'OL') to the annotator segmentations. Despite the mismatch, since there is a similar behavior for all three ('NUM', 'CI', 'OL'), the ratio of numerator to denominator is close to 1. In the utterance "monkey" shown in 9b, the 'CI' element failed to recognized "N" phoneme, and the beginning of its divergence occurred at the end of the first phoneme "SIL." This initial change led to different Viterbi



(a) correct-label detection (house)

(b) CI misclassification (monkey)

(c) OL misclassification (fishing)

(d) incorrect-label detection (ring)

Figure 9: Different recognition results along with human annotator decisions

decisions resulting in a different path for 'CI'. On the other hand, when describing the word "fishing," in Figure 9c there is a mis-recognition of 'OL' with "F" and "S". Another issue occurred in both 'CI' and 'OL' when recognizing "N", which was confused with a distorted version of "N". The last example, shown in Figure 9d, demonstrates the expected behavior when a phoneme is mispronounced. The word is "ring", yet the child, according to the expert, uttered "wing". The numerator can only present a correct version of phonemes, thus the choice of "9R" is forced, yet both 'CI' and 'OL' detected that "W" was in fact pronounced. This set of case studies strengthens our assumptions regarding the output of the system while applying the proposed methods to our data.

5.2. Recall Test

In the second part of the data analysis, we conducted recall tests among GOP-CI presented in Eq. 14, GOP-SVM presented in Section 4.1.2, and LGOP described in Eq. 7. Each system's decision was compared against a human expert decision who annotated the CCP dataset and the recall test provides the agreement rate of a system with the human expert. There were several criteria that measured the success of each method:

Total Recall -

$$TotR = \left(\frac{Sys_P \bigcap Ex_P}{Ex_P} \right) \tag{18}$$

Label Recall -

$$C = (\frac{Sys_{P_C} \bigcap Ex_{P_C}}{Ex_{P_C}}), I = (\frac{Sys_{P_I} \bigcap Ex_{P_I}}{Ex_{P_I}}) \tag{19}$$

Unweighted Average Recall -

$$UAR = 0.5 * C + 0.5 * I \tag{20}$$

$Sys_{P_*} \bigcap Ex_{P_*}$ refers to positive results, which both the system and the expert agreed upon. Sys_{P_C}, Sys_{P_I}, and Sys_P refer to positive results (true positive) of system's recognition of 'C' labels, 'I' labels, and all labels respectively. 'C' label refers to a 'correct pronunciation' while 'I' label refers to an 'incorrect' pronunciation[1]. UAR is the Unweighted Average Recall rate that was used to overcome the disproportional distribution of data favoring 'C' label. 75% of the data contained 'C' labels whereas 25% were 'I' labels. Thus, while Total recall is highly affected by the performance of 'C' labels recall rate, the UAR presents an equal weight for each label and therefore is evenly affected by both labels' recall rate. A Random classifier provided the lower bound results for the three presented methods. It learned to randomly classify results proportionally to the

[1]to recall how each method decides whether a phoneme is correctly pronounced or incorrectly pronounced the reader can refer to the equations and explanations of these approaches mentioned at the beginning of this subsection.

distribution of classes found in the data. (in a stratified fashion). The Random classifier was created with Sickit-learn toolkit using a "dummy classifier" [37].

Table 2 indicates the results of the Recall test for the five-fold cross validation experiment. Each outcome from Eq. 19, 20, and 18 was multiplied by 100 to represent a percentage. All methods' parameter training processes were set with the objective of maximizing the UAR.

	fold	$TotR$	C	I	UAR
GOP-CI		64.70	65.36	60.61	62.98
GOP-SVM	1	60.47	67.04	59.41	63.22
LGOP		50.08	46.97	69.32	58.14
Random		73.03	82.21	16.28	49.25
GOP-CI		65.09	65.67	61.09	63.38
GOP-SVM	2	63.54	64.43	63.40	63.92
LGOP		52.61	50.21	69.04	59.63
Random		74.04	82.34	17.15	49.75
GOP-CI		58.94	58.93	59.05	58.99
GOP-SVM	3	56.25	65.76	55.16	59.96
LGOP		47.63	45.50	64.29	54.89
Random		74.73	82.26	15.71	48.98
GOP-CI		68.15	68.32	67.16	67.74
GOP-SVM	4	63.38	72.32	61.90	67.11
LGOP		51.81	48.66	70.85	59.75
Random		73.03	82.21	16.28	49.25
GOP-CI		67.88	67.78	68.34	68.06
GOP-SVM	5	64.60	71.89	63.05	67.47
LGOP		52.70	48.68	71.60	60.14
Random		71.56	82.84	18.63	50.74

Table 2: Recall results over 5-fold testsets

A first look at the results indicates that the Random classifier has the lowest UAR and hence the Random classifier provides the lower bound of results for all other approaches. Because of the disproportional distribution of the data, the Random classifier guessed 'C' label more often, explaining its high recall rate and conversely, the low recall rate for 'I' label. Moreover, Table 2 reveals that LGOP has a lower performance in UAR than GOP-SVM and GOP-CI. In addition, the total recall is higher for both methods (GOP-SVM and GOP-CI) across all folds. However, when inspecting the recall rate of labels 'C' and 'I', it is not clear whether our methods always achieved optimal results. While the GOP-CI-based method and GOP-SVM recognize 'C' labels distinctly well, they under-perform with 'I' labels. The recall rates for 'I' of GOP-CI and GOP-SVM are suboptimal, though they are better than chance and relatively closer to LGOP scores. On the other hand, the LGOP's 'C' recall rates are below chance and at a greater distance from the 'C's of GOP-CI and GOP-SVM. In other words, the three algorithms were relatively close in identifying cases where the child pronounced the phoneme incorrectly (i.e., 'I' labels), while cases where the child pronounced a phoneme correctly (i.e., 'C' labels) were identified more often by the GOP-CI and GOP-SVM than LGOP. These observations explain why average UAR is higher in the proposed methods than in LGOP.

Between the two proposed methods, it is more challenging to draw definite conclusions regarding which is more optimal. On the one hand, GOP-SVM performed better with 'C' labels recognition, while on the other, GOP-CI made better decisions regarding 'I' labels. Table 3 is summarizing the five-fold results of Table 2 (using simple averaging).

	$TotR$	C	I	UAR
GOP-CI	64.95	65.21	63.25	64.23
GOP-SVM	61.64	68.30	60.58	64.33
LGOP	51.00	48.00	69.02	58.51

Table 3: Average results over 5-folds

While LGOP in Table 3 classifies 'I' labels relatively well, it poorly classifies 'C' labels. Moreover, both UAR and total recall of LGOP are below the scores of our proposed methods. When comparing GOP-SVM to GOP-CI in Table 3, there is only a small difference (0.1%-3%) between the two approaches (in 'C', 'I', UAR and total recall). However, when examining closely individual performance of 'C' and 'I' recognition rates of both methods, 'C' and 'I' are recognized at a similar rate by GOP-CI while GOP-SVM demonstrates a greater gap between the labels' recognition recall. Arguably from a developer's point of view, GOP-CI would be preferred as there would be less bias towards a particular label. However, the investigation continues and a statistical test (in section 5.3) is applied to further examine these methods with different tools. These subsections together provide a more in-depth picture of the different methods' performance.

5.3. Statistical Test

5.3.1. Model and Variables for Testing

The fact that our data contained repeated measurements (i.e., multiple productions of the same phoneme by the same child) complicated our modeling process, and we ultimately chose to use Generalized Linear Mixed Effect Modeling (GLMM) [39]. GLMM allowed us to carefully examine the differences stemming solely from the use of LGOP, GOP-SVM and GOP-CI methods, while holding constant the other variables of the data such as the child, their age, the phoneme uttered and the word produced. The specific GLMM model was Logistic Regression, as our data had a binary response variable (dependent variable): whether the automated pronunciation method agreed with the expert on a decision about a specific phoneme. Saying that a given method of pronunciation analysis "agreed with the expert" in this case means that both the method and expert *agreed* that a particular phoneme was correctly pronounced (or alternatively, that it was incorrectly pronounced).

Therefore, the question our model is asking is whether there is a difference between LGOP, GOP-SVM and GOP-CI methods in the probability of a given method *agreeing* with the expert. The assumption is that a 'better' system is

one that has a higher probability for agreeing with the expert. GLMM modeling lets us examine and assess each variable independently. In particular, one of the model's outputs is each variables' weights (coefficients), as the model is a linear combination of the variables with their corresponding weights. This allows us to examine the relative impact on agreement of the various model parameters. This section first presents the model, then describes the test performed and finally analyzes the results.

The GLMM model had both random and fixed effects. In the mixed effect model, random effects are defined as elements that have unpredictable influence on the data, and that might contain multiple observations for the subgroup of the element. The random effects were:

subject - The data contains multiple observations for each subject. Moreover, there is no predictability to how often each subject will correctly utter a phoneme since every child (subject) might experience different pronunciation difficulties depending on the word and phoneme. It is important to note that while among the children there is no predictability, within a child there can be observed a systematic behavior which is taken into account when modeling the data.

phoneme - The data contains multiple observations for each phoneme. In addition, there is no predictability with regard to how often a phoneme is correctly pronounced.

word - A word is a phonemic sequence. Thus, its effect is similar in concept to the *phoneme* effect, since it is challenging to predict how well a phoneme is pronounced in the presence of adjacent phonemes. Different phonemic contexts represented by the *word* variable may affect pronunciation quality. The data contains multiple observations for each word uttered by each subject.

The fixed effects, defined as predictable or more systematic than the random effects, were:

age - The older a child gets, the more often on average we expect her utterances are correctly pronounced.

method - A categorical variable to represent each of the three automatic pronunciation systems: GOP-CI, GOP-SVM, and LGOP. Every system is assumed to be correlated to some degree with the expert decisions and as a result tends to agree with the expert. Therefore, this variable is considered as a predicted fixed effect. "Indicator Variables" [40] were used for the coding system described in each experiment.

All the variables above are the model components that try to explain the *hit* variable.

hit - A variable representing *hit* values corresponds to an agreement (or disagreement) of the method with the expert's decision. For instance, if both a method and the expert determined a phoneme to be correctly pronounced then it is a 'hit', (coded for modeling purposes as "1"). Conversely, if a method decided that a phoneme was incorrectly pronounced while the expert considered the phoneme to be correctly pronounced, then it is not a hit, and is coded 0 in the *hit* variable.

The GLMM model that was built is described in Equation 21:

$$mdl = hit \sim (1|subject) + (1|phoneme) + (1|word) + age + method \qquad (21)$$

We used "lme4" package [41] found in "R", which is the programming language and software applied to create the model and run the following statistical tests. One possible limitation when performing the following statistical analysis could be caused by the number of samples of each fold of the five-fold cross validation test sets and number of subjects it contained. The sample size ranged from 1697-1897 and the number of subjects ranged between 8-9 subjects in a fold.

5.3.2. Testing for the Significance of the Coefficients

The outcome of the model in Equation 21 provided estimated coefficients for each of the independent variables. The variable of interest in our experiments was *method* representing which automated pronunciation system was used. The question at hand in the first experiment was whether the different pronunciation methods differed from one another in their probability of agreement with the human expert, after controlling for all variables other than *method*. Specifically in GLMM, this question corresponds to the question of whether the model coefficients for the *method* variable were significantly different from zero. The model in Equation 21 incorporated the coding system of Table 4 for each pronunciation system.

LGOP	0	0
GOP-SVM	1	0
GOP-CI	0	1

Table 4: First Coding System

In this coding system, each level of *method* (GOP-CI and GOP-SVM) is compared to the reference level of LGOP. As a result of using this coding system, the outcome coefficients of the model indicate the difference between LGOP and GOP-CI, and LGOP and GOP-SVM, called β_1 and β_2 respectively. Since β_1 and β_2 describe the differences between LGOP to GOP-SVM and LGOP to GOP-CI, significant differences, which are values that are significantly different than zero, demonstrate that a pronunciation system (GOP-SVM or GOP-CI) is associated with a different probability (than LGOP) for agreement with the expert.

To measure whether the coefficients of *method* variable are significant, the Wald Test [42] was applied. The H_0 (null hypothesis) was that the estimated coefficient has no effect on the dependent variable *hit* and is not associated with a different probability for method-expert agreement. On the other hand, H_1 was

that the estimated coefficient is important to explain *hit* and differs for LGOP and the method, indicating that this coefficient is associated with a different probability (than LGOP) for method-expert agreement.

The test statistic is in Equation 22[2]:

$$z^* = \frac{\hat{\beta}_k}{var(\hat{\beta}_k)} \tag{22}$$

and the decision rule is in Equation 23:

$$\begin{aligned} if \quad |z^*| \leq z(1-\alpha/2), \quad conclude \quad H_0 \\ if \quad |z^*| > z(1-\alpha/2), \quad conclude \quad H_1 \end{aligned} \tag{23}$$

For $\alpha = 0.025$ and one degree of freedom the decision threshold required $z(0.975) = 1.960$. Table 5 demonstrates the z estimates for βs in the five-fold cross validation experiment previously described in Section 5.2.

	fold	est.	z^*	OR
$\hat{\beta}_1$	1	0.44	6.575	1.56 (1.37-1.78)
$\hat{\beta}_2$		0.64	9.313	1.89 (1.65-2.16)
$\hat{\beta}_1$	2	0.48	6.987	1.61 (1.41-1.85)
$\hat{\beta}_2$		0.55	8.022	1.74 (1.52-1.99)
$\hat{\beta}_1$	3	0.36	5.394	1.44 (1.26-1.64)
$\hat{\beta}_2$		0.49	7.147	1.62 (1.42-1.86)
$\hat{\beta}_1$	4	0.49	6.904	1.64 (1.43-1.89)
$\hat{\beta}_2$		0.73	9.972	2.07 (1.80-2.50)
$\hat{\beta}_1$	5	0.55	7.626	1.73 (1.50-2.00)
$\hat{\beta}_2$		0.70	9.660	2.02 (1.76-2.34)

Table 5: Wald Test for Coefficient Significance (1).

As seen in Table 5, all the coefficients indicate that H_0 is rejected since all β estimates in Table 5 are greater than $z = 1.960$. The conclusion resulting from

[2] $\hat{\beta}$ is the estimate provided by the model for the true coefficient β

rejecting H_0 is that in every fold examined, after controlling for all variables other than *method*, GOP-SVM and GOP-CI were associated with different (than LGOP) probability of agreeing with the expert. This conclusion strengthens the previous results in the Recall test in Section 5.2, in which it was shown that both GOP-SVM and GOP-CI achieved higher (different) total recall rates than LGOP. The last column in Table 5 is of odds ratio (OR) [40]. In a Logistic Regression model OR can be retrieved by exponentiating the estimated βs to reveals the odds of a method to have a higher probability (than LGOP) for agreeing with the expert.

As shown in table 5, the point estimates for β_1 from the five-fold cross-validation ranges from 1.44 to 1.73, with an average of 1.6. This means that, on average, the odds of GOP-SVM producing hits are roughly 1.6 times those of LGOP producing a hit, after controlling for the effects of subject, age, phoneme, and word. Similarly, all β_2 point estimates have an odd ratio range of 1.62-2.07 with an average of 1.85, which means that on average the odds of GOP-CI of producing *hits* are almost 1.85 times as great as for LGOP, for given child, age, phoneme, and word. The OR findings indicate that there is a higher probability for agreeing with the expert associated with GOP-SVM and GOP-CI. The 95% confidence intervals are provided in OR column in parenthesis and show that for all βs, none of these intervals includes the value of one, which means that the βs are significantly different than zero (as exponentiating zero results in one).

We next sought to answer a slightly different question: that of whether, after controlling for all other variables other than *method*, there is a difference between GOP-SVM and GOP-CI. The model in Equation 21 incorporated the coding system of Table 6 for each pronunciation system.

GOP-SVM	0	0
GOP-CI	1	0
LGOP	0	1

Table 6: Second Coding System

In this coding system each level *method* (GOP-CI and LGOP) is compared to the reference level of GOP-SVM. As a result of using this coding system, the outcome coefficients of the new model indicate the difference between GOP-SVM and GOP-CI, and GOP-SVM and LGOP, called β_1 and β_2 respectively. Accordingly, the purpose of the second experiment is to measure the differences between GOP-SVM and GOP-CI. The H_0 was similar in concept to one asked in the first experiment. Table 7 demonstrates the results for the same five-fold cross validation sets.

	fold	est.	z^*	OR
$\hat{\beta_1}$	1	0.19	2.795	1.21 (1.06-1.39)
$\hat{\beta_1}$	2	0.07	1.055	1.08 (0.94-1.23)
$\hat{\beta_1}$	3	0.12	1.775	1.13 (0.98-1.29)
$\hat{\beta_1}$	4	0.23	3.153	1.26 (1.09-1.46)
$\hat{\beta_1}$	5	0.16	2.097	1.17 (1.01-1.35)

Table 7: Wald Test for Coefficient Significance (2).

As seen in Table 7, β_1 coefficients indicate that H_0 is rejected in folds 1, 4, and 5, while H_0 is not rejected in folds 2 and 3 as these folds' z estimates fall below $z = 1.960$. Due to these conflicting results as β_1 estimates were not greater then their z estimates across all folds the conclusion is that the outcome of GOP-SVM and GOP-CI comparison was not always associated with different probability of agreeing with the expert. This conclusion strengthens the previous results in Recall test in (Section 5.2), in which it was shown that both GOP-SVM and GOP-CI shared similar total recall rates that accounted for slightly different label recall rates. The odds ratios (OR) of GOP-CI, though, seem to be consistently greater than GOP-SVM indicating a higher probability for GOP-CI (than GOP-SVM) for agreeing with the expert. As shown in Table 7, all β_1 point estimates have an odd ratio that ranges between 1.08-1.26 with an average of 1.17, which means that on average the odds of GOP-CI of

producing hits are almost 1.17 times as great as for GOP-SVM, for given child, age, phoneme, and word. The OR findings indicate that there is a higher probability associated with GOP-CI compared with GOP-SVM for agreeing with the expert. However, since about half the folds rejected and half the folds accepted the H_0, the differences between the methods remained insignificant. This finding indicates that, for folds 2 and 3, we were unable to be certain at a 95% level of confidence that the true value for β_1 was greater than zero. Additionally, the confidence intervals of folds 1, 4 and 5 are not as distant from one either, which might decrease the certainty in these folds' of being different than zero as well.

In this section we provided an additional test that examined the methods after controlling for all other variables such as age, word, phoneme, and child to measure whether a particular method has a higher probability for agreeing with the expert. Clearly, the statistical analysis emphasized that GOP-SVM and GOP-CI provide more valuable models, as they are highly more likely to agree with the expert decisions (as shown in the first experiment in this section). In addition, the GOP-SVM and GOP-CI examination held in the second experiment showed that while GOP-CI consistently indicated a higher probability for agreeing with the expert, no significant differences were observed between the two methods.

5.4. Error Analysis Test

In addition to conducting the analysis in Section 5.2 and the statistical tests in Section 5.3 we wished to better characterize the differences between GOP-CI to GOP-SVM. The previous sections demonstrated that LGOP undoubtedly under-performs GOP-CI and GOP-SVM, yet there was an incomplete explanation for the reason for which both methods slightly differ in performance. In order to answer this final question we conducted an error analysis test that measures how many relevant label assignments are selected (recall), how many of the selected label assignments are relevant (precision), and a score to summarize recall and precision. The following analysis provides yet additional way to examine the models' performance by focusing on "false negative" (through

recall) "false positive" rate (through precision).

Table 8 demonstrates the methods' performance measured by precision, recall, and $f1$ (called also harmonic mean) metrics by calculating the metrics' "macro" average. "Macro" average computes the average over the different classes for recall, precision, and $f1$ score (as oppose to "micro" average). Therefore, the final $f1$ score is not a harmonic mean of the precision and recall found in Table 8, rather it is a mean of the $f1$ scores of 'C' ('correct') and 'I' ('incorrect') classes. Recall is computed as presented in Eq. 20 and similarly, precision is also an unweighted average of precision of the 'C' and 'I' labels.

	fold	*Precision*	*Recall*	$f1$ score
GOP-CI		0.56	0.63	0.54
GOP-SVM	1	0.56	0.63	0.52
LGOP		0.53	0.58	0.45
GOP-CI		0.56	0.63	0.54
GOP-SVM	2	0.56	0.63	0.53
LGOP		0.54	0.59	0.46
GOP-CI		0.53	0.59	0.48
GOP-SVM	3	0.54	0.60	0.47
LGOP		0.52	0.55	0.41
GOP-CI		0.59	0.67	0.58
GOP-SVM	4	0.58	0.67	0.55
LGOP		0.55	0.60	0.46
GOP-CI		0.61	0.68	0.60
GOP-SVM	5	0.60	0.67	0.58
LGOP		0.56	0.60	0.49

Table 8: Precision, Recall, and $f1$-score for 5-fold testsets

While precision scores in Table 8 indicate similar performance among the three methods, recall scores may be similar only for GOP-SVM and GOP-

CI with a noticeable gap between them compared with LGOP. According to Table 8, in all folds, GOP-CI had a higher $f1$ score than GOP-SVM. One explanation for this outcome requires a closer examination of each class recall results provided in Table 9.

	label	*Precision*	*Recall*	$f1$ score
GOP-CI	C	0.91	0.65	0.76
GOP-CI	I	0.22	0.61	0.32
GOP-SVM	C	0.92	0.59	0.72
GOP-SVM	I	0.21	0.67	0.32

Table 9: A typical example (taken from first fold) for labels' precision, recall and $f1$ score

One example to further explain the table's cells is to search for the cell on the third column from the left that intersects the second row. This cell with a value of 0.91 describes the precision rate for 'C' label for the GOP-CI.

GOP-CI's recall rate was high for 'C' label (0.65) and since it had high precision score (0.91) as well (and similar to GOP-SVM) its final $f1$ score was relatively higher than GOP-SVM. On the other hand, recall on 'I' label was better on GOP-SVM (0.67), however its impact on $f1$ was small due to low (and similar to GOP-CI) precision rate (0.21). Therefore, both methods produced similar $f1$ scores for 'I' label while GOP-CI had greater $f1$ value for 'C' label. In other words, since $f1$ is a harmonic mean of precision and recall, it is necessary for both measures to represent high values in order to have a high $f1$ score. Thus, GOP-CI had greater overall $f1$ scores because while $f1$ score for 'I' was similar for both methods, GOP-CI had higher $f1$ score for 'C' label. Thus, averaging the scores for each method resulted in a higher overall $f1$ score for GOP-CI. This piece of evidence supports the statistical test results and explains the recall test in greater depth. While it was essential to discuss the differences between GOP-CI and GOP-SVM the overall scores are of the main interest. Therefore, the presented scores emphasize that GOP-CI and similarly GOP-

SVM are highly probable to be effective models in practice.

6. Discussion and Conclusions

In the current paper we proposed two automatic decision methods, GOP-CI and GOP-SVM, aimed at analyzing children's speech for children who may be facing with speech-sound disorders. To evaluate the proposed methods, the most recent state of the art GOP - LGOP served as an automatic method of reference.

The first proposed method was the GOP-CI whose parameters were set using a grid search. The second proposed method was GOP-SVM which was trained using an SVM approach. Both methods learned patterns found in the CCP corpus that contained correct and incorrect pronunciations of children with speech-sound disorders. This unique dataset provided authentic recordings of children along with corresponding human expert annotations. While the acoustic recordings were added to the ASR training, the collection of annotations determined the sets of phoneme candidates of 'correct' and 'incorrect' groups. The expert's decisions of 'correctly pronounced' or 'incorrectly pronounced' phonemes were the gold standard in our experiment.

After having been produced for each approach, the models were examined from different perspectives. First, the recall test demonstrated that the UAR (Unweighted Average Recall) as well as the total recall were greater for the proposed methods than the LGOP. In addition, when comparing the GOP-CI to GOP-SVM, arguably, GOP-CI performed slightly better for providing a more balanced recognition rate for each label. Second, the statistical test indicated that the GOP-CI and GOP-SVM were associated with a higher (than LGOP) probability for agreeing with the expert. Additionally, while GOP-CI showed a higher probability than GOP-SVM for method-expert agreement, the overall results showed that these methods were not statistically different. Lastly, the $f1$ score was computed for each method. The scores indicated a small gain for GOP-CI relative to GOP-SVM and a low $f1$ score for LGOP.

It appears that the reason that the GOP-CI and GOP-SVM outperformed the baseline approach is that both approaches incorporated additional features of the different classes found in the data. For each context-dependent phoneme, GOP-CI and GOP-SVM were provided with their incorrect phoneme group while the nature of LGOP did not discriminate any phoneme in its process. The GOP-CI had marginally better performance than GOP-SVM. However, fine tuning of hyper parameters of the SVM could have shown similar results on the upper bound of the GOP-CI-based approach. In conclusion, GOP-CI and GOP-SVM have proven that they have better performance than the baseline of LGOP. Therefore, it appears that progress has been made towards an improved automatic speech analysis tool.

7. Future Work

In order to create more robust models, one of our future plans is to improve the current state-of-the art for annotation scores. This could be done by adding an increased number of human expert annotators, producing a model that would learn a pattern according to a voting system and be able to reflect several experts decisions. However, this should take into account the level of disagreement rate among the annotators, since high levels of disagreement would produce an ineffective tool for evaluating pronunciation performance.

Another future enhancement would be to improve utterance phoneme segmentation in deletion and insertion events. Deletion and insertion events may shift the phoneme sequence relatively to the expected phoneme sequence of a word which may prevent the system from comparing the uttered phoneme to the expected one. Thus, to avoid a flawed comparison the uttered word should be aligned with respect to the expected word. Alignment requires a comparison of the pronounced word output of the ASR with the expected word. In our experiment, deletion and insertion events that were at the end a word presented no issue while other positions of deletion and insertion were excluded from our data. While those excluded utterances were less than 2% in CCP

corpus, adding a comparison process would enable the systems to accurately process any possible utterance with deleted phonemes or inserted ones.

We believe that by applying these improvements to our proposed methods a further step can be taken towards a more reliable analysis of children's speech.

8. Acknowledgements

Funding: The presented research was supported by the National Institutes of Health [grant number R21DC012139].

9. References

[1] J. Gierut, Treatment Efficacy: Functional Phonological Disorders in Children, Journal of Speech, Language, and Hearing Research 41 (1998) S85–S100.

[2] A. Castrogiovanni, Incidence and Prevalence of Communication Disorders and Hearing Loss in Children, American Speech-Language Hearing Association, 2008.

[3] K. Squires, Addressing the Shortage of Speech-Language Pathologists in School Settings, Journal of the American Academy of Special Education Professionals (2013).

[4] X. Huang, Y. Ariki, M. Jack, Hidden Markov Models for Speech Recognition, volume 2004, Edinburgh university press Edinburgh, 1990.

[5] S. Witt, S. Young, Computer-assisted pronunciation teaching based on automatic speech recognition, Language Teaching and Language Technology Groningen, The Netherlands (1997).

[6] A. Neri, C. Cucchiarini, H. Strik, L. Boves, The Pedagogy-Technology Interface in Computer Assisted Pronunciation Training, Computer Assisted Language Learning 15 (2002) 441–467.

[7] A. Neri, O. Mich, M. Gerosa, D. Giuliani, The Effectiveness of Computer Assisted Pronunciation Training for Foreign Language Learning by Children, Computer Assisted Language Learning 21 (2008) 393–408.

[8] T. Bunnell, D. Yarrington, J. Polikoff, STAR: articulation training for young children, in: INTERSPEECH, the Proceedings of the Conference, Beijing, China, IEEE, 2000, pp. 85–88.

[9] M. Russell, C. Brown, A. Skilling, R. Series, J. Wallace, B. Bonham, P. Barker, Applications of Automatic Speech Recognition to Speech and Language Development in Young Children, in: ICSLP Proceedings., Fourth

International Conference, Philadelphia, USA, volume 1, IEEE, 1996, pp. 176–179.

[10] M. Eskenazi, S. Hansma, The FLUENCY pronunciation trainer, in: Proceedings of the STiLL Workshop, 1998.

[11] G. Kawai, K. Hirose, A CALL System Using Speech Recognition to Train the Pronunciation of Japanese Long Vowels, the Mora Nasal and Mora Obstruents, in: EUROSPEECH 1997, Rhodes, Greece, September 22-25, 1997, 1997.

[12] F. Ehsani, E. Knodt, Speech Technology in Computer-Aided Language Learning: Strengths and Limitations of a New CALL Paradigm, Language Learning & Technology 2 (1998) 45–60.

[13] L. Koenig, Distributional Characteristics of VOT in Children's Voiceless Aspirated Stops and Interpretation of Developmental Trends, Journal of Speech, Language & Hearing Research 44 (2001).

[14] M. Benzeghiba, R. De Mori, O. Deroo, S. Dupont, T. Erbes, D. Jouvet, L. Fissore, P. Laface, A. Mertins, C. Ris, et al., Automatic Speech Recognition and Speech Variability: A Review, Speech Communication 49 (2007) 763–786.

[15] B. Ploog, A. Scharf, D. Nelson, P. Brooks, Use of Computer-Assisted Technologies (CAT) to Enhance Social, Communicative, and Language Development in Children with Autism Spectrum Disorders, Journal of Autism and Developmental Disorders 43 (2013) 301–322.

[16] V. J. Doremalen, C. Cucchiarini, H. Strik, Automatic Pronunciation Error Detection in Non-Native Speech: The Case of Vowel Errors in Dutch, The Journal of the Acoustical Society of America 134 (2013) 1336–1347.

[17] S. Ronanki, J. Salsman, T. Li, Automatic Pronunciation Evaluation and Mispronunciation Detection Using CMUSphinx, in: 24th International Conference on Computational Linguistics, Citeseer, 2012, p. 61.

[18] S. Wei, G. Hu, Y. Hu, R. H. Wang, A New Method for Mispronunciation Detection Using Support Vector Machine Based on Pronunciation Space Models, Speech Communication 51 (2009) 896–905.

[19] H. Strik, K. Truong, F. De Wet, C. Cucchiarini, Comparing Different Approaches for Automatic Pronunciation Error Detection, Speech Communication 51 (2009) 845–852.

[20] S. Witt, S. Young, Phone-level Pronunciation Scoring and Assessment for Interactive Language Learning, Speech Communication 30 (2000) 95–108.

[21] B. Mak, M. Siu, M. Ng, Y. C. Tam, Y. C. Chan, K. W. Chan, K. Y. Leung, S. Ho, F. H. Chong, J. Wong, PLASER: Pronunciation Learning via Automatic Speech Recognition, in: Proceedings of the HLT-NAACL workshop on Building educational applications using natural language processing-Volume 2, Association for Computational Linguistics, 2003, pp. 23–29.

[22] Y. Song, W. Liang, R. Liu, Lattice-Based GOP in Automatic Pronunciation Evaluation, in: Computer and Automation Engineering (ICCAE), 2010 The 2nd International Conference on, volume 3, IEEE, 2010, pp. 598–602.

[23] A. Viterbi, Error Bounds for Convolutional Codes and an Asymptotically Optimum Decoding Algorithm, Information Theory, IEEE Transactions on 13 (1967) 260–269.

[24] P. Swain, H. Hauska, The Decision Tree Classifier: Design and Potential, Geoscience Electronics, IEEE Transactions on 15 (1977) 142–147.

[25] M. A. Peabody, Methods for Pronunciation Assessment in Computer Aided Language Learning, Ph.D. thesis, Massachusetts Institute of Technology, 2011.

[26] K. Zechner, D. Higgins, X. Xi, D. M. Williamson, Automatic Scoring of Non-Native Spontaneous Speech in Tests of Spoken English, Speech Communication 51 (2009) 883–895.

[27] N. Minematsu, Pronunciation Assessment Based Upon the Compatibility Between a Learner's Pronunciation Structure and the Target Language's Lexical Structure, in: INTERSPEECH 2004, ICSLP, 8th International Conference on Spoken Language Processing, Jeju Island, Korea, October 4-8, 2004.

[28] C. Cortes, V. Vapnik, Support Vector Machine, Machine learning 20 (1995) 273–297.

[29] L. Weigelt, S. Sadoff, J. Miller, Plosive/Fricative Distinction: The Voiceless Case, The Journal of the Acoustical Society of America 87 (1990) 2729–2737.

[30] R. Goldman, M. Fristoe, Goldman-Fristoe test of articulation-2, 2000.

[31] B. Paul, Praat, a System for Doing Phonetics by Computer, Glot international 5 (2002) 341–345.

[32] A. B. Smit, Phonologic Error Distributions in the Iowa-Nebraska Articulation Norms Project Consonant Singletons, Journal of Speech, Language, and Hearing Research 36 (1993) 533–547.

[33] L. Graham, A. House, Phonological Oppositions in Children: A Perceptual Study, The Journal of the Acoustical Society of America 49 (1971) 559–566.

[34] S. Dudy, M. Asgari, A. Kain, Pronunciation Analysis for Children with Speech Sound Disorders, in: Engineering in Medicine and Biology Society (EMBC), 2015 37th Annual International Conference of the IEEE, IEEE, 2015, pp. 5573–5576.

[35] D. Povey, A. Ghoshal, G. Boulianne, L. Burget, O. Glembek, N. Goel, M. Hannemann, P. Motlicek, Y. Qian, P. Schwarz, J. Silovsky, G. Stemmer, K. Vesely, The Kaldi Speech Recognition Toolkit, in: IEEE 2011 Workshop on Automatic Speech Recognition and Understanding, IEEE Signal Processing Society, 2011. IEEE Catalog No.: CFP11SRW-USB.

[36] C. Allauzen, M. Riley, J. Schalkwyk, W. Skut, M. Mohri, OpenFst: a General and Efficient Weighted Finite-State Transducer Library, in: Proceedings of the Ninth International Conference on Implementation and Application of Automata, (CIAA 2007), volume 4783 of *Lecture Notes in Computer Science*, Springer, 2007, pp. 11–23. `http://www.openfst.org`.

[37] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, et al., Scikit-learn: Machine Learning in Python, Journal of Machine Learning Research 12 (2011) 2825–2830.

[38] K. Shobaki, J.-P. Hosom, R. Cole, The OGI Kids' Speech Corpus and Recognizers, in: ICSLP, the Proceedings of the Conference, Beijing, China, 2000, pp. 564–567.

[39] C. McCulloch, J. Neuhaus, Generalized Linear Mixed Models, Wiley Online Library, 2001.

[40] M. Kutner, C. Nachtsheim, J. Neter, W. Li, Applied Linear Statistical Models, McGraw-Hill Irwin New York, 2005.

[41] D. Bates, M. Mächler, B. Bolker, S. Walker, Fitting Linear Mixed-Effects Models Using lme4, Journal of Statistical Software 67 (2015) 1–48.

[42] A. Buse, The Likelihood Ratio, Wald, and Lagrange Multiplier Tests: An expository note, The American Statistician 36 (1982) 153–157.