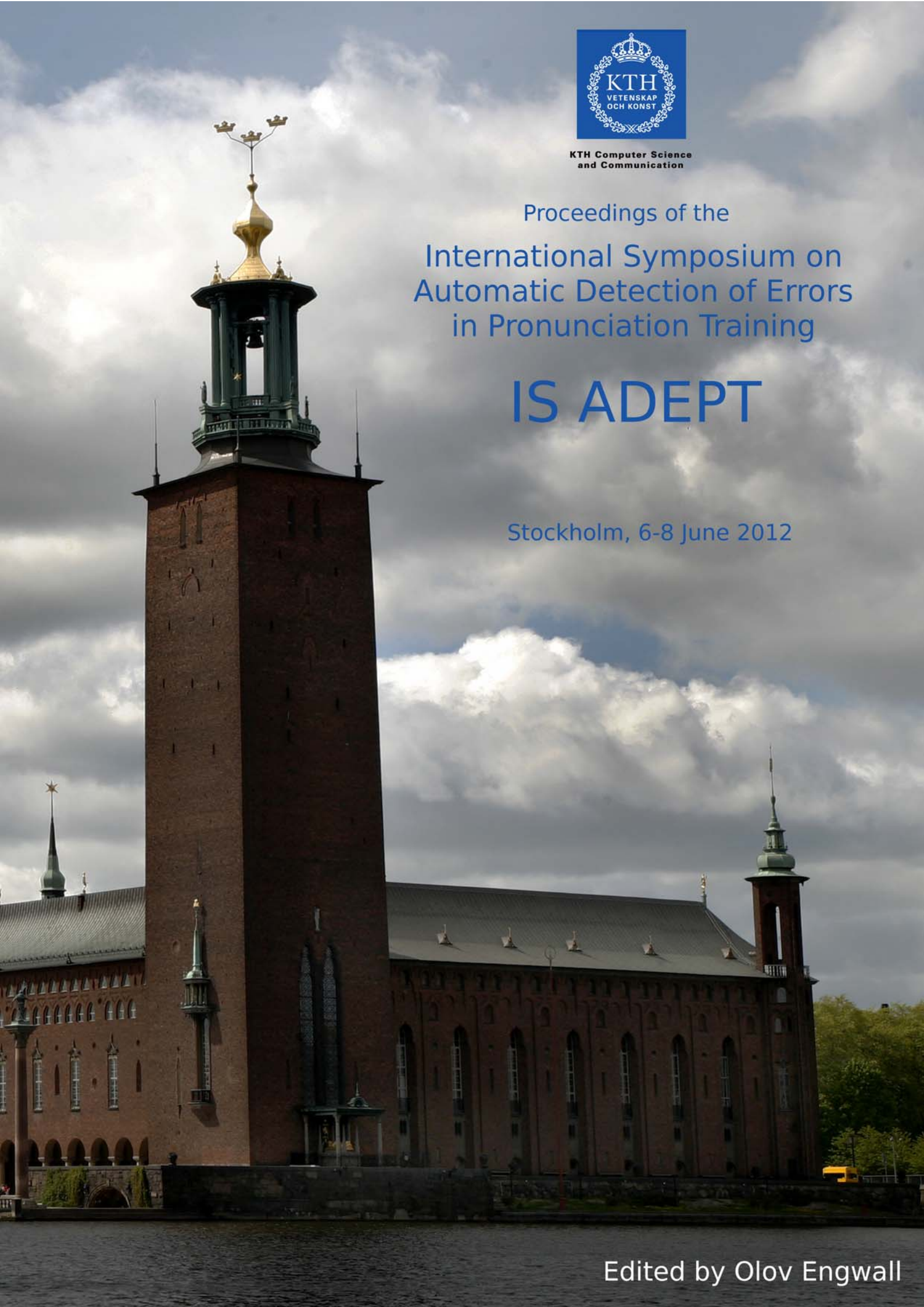


KTH Computer Science and Communication

Proceedings of the

International Symposium on Automatic Detection of Errors in Pronunciation Training

IS ADEPT

Stockholm, 6-8 June 2012

Edited by Olov Engwall

KTH Computer Science
and Communication

Proceedings of the

International Symposium on Automatic Detection of Errors in Pronunciation Training

June 6 – 8, 2012

KTH, Stockholm, Sweden

In collaboration with:

Cairo University, Egypt

Sponsored by:

Vetenskapsrådet

Proceedings of the
International Symposium on Automatic Detection of Errors in Pronunciation Training

Editor: Olov Engwall

ISBN: 978-91-7501-402-9

Published by
KTH, Computer Science and Communication
Department of Speech, Music and Hearing
SE-100 44 Stockholm, Sweden

Electronic version

Cover photo by Yanan Li
LaTeX definitions for proceedings by Giampiero Salvi
(`https://github.com/giampierosalvi/LaTeXProceedings`)

Foreword

Welcome to the International Symposium on Automatic Detection of Errors in Pronunciation Training (IS ADEPT). Fourteen years ago, the first European Speech Communication Association (ESCA) workshop on Speech Technology in Language Learning (STiLL) was organized at Marholmen, Sweden by KTH (Royal Institute of Technology). We are very happy to once again welcome the computer assisted pronunciation training (CAPT) community to Stockholm, at a symposium endorsed by SLaTE, the ISCA special interest group on Speech and Language Technology in Education, and ALTEC, the Arabic Language Technology Center.

The aim of this symposium is to bring together academia and industry in the CAPT field to promote the exchange of ideas on the current state-of-the-art for automatic pronunciation analysis, and on needs and paths for future research and development. The main part of the symposium program consists of eight invited speakers, each covering an important aspect of the field.

Academic research is represented by Silke Witt, Helmer Strik, Florian Hoenig and Jack Mostow, who will give an overview of past and current trends in automatic pronunciation analysis at the phonemic and prosodic level, in read or spontaneous speech.

The CAPT industry is represented by Lewis Johnson, Bryan Pellom and Gary Pelton, who each will describe the use of automatic pronunciation training in the systems that their companies are developing. These industrial presentations range from a focus on the detection of specific mispronunciations to general system descriptions and hence cover several different levels of practicing with CAPT systems.

The final keynote speaker, Horacio Franco, represents the border between academic research and industrial development, a border that several of the keynote speakers have in fact traversed, and that this symposium aims at making more active and fruitful.

Silke Witt is Vice President of Speech Solutions at Fluential Inc, where she is working on building the next generation multi-modal dialog systems. Her keynote paper is a review of the research on automatic pronunciation error detection that has been conducted over the past 10-15 years. Silke Witt is ideally placed to perform such a review, given her long experience of academic and industrial research on speech technology, starting with her work on the use of ASR in computer-assisted language learning in the late 1990's. This work, performed in collaboration with Steve Young, remain some of the most cited articles on the topic.

Helmer Strik, Centre for Language and Speech Technology at Radboud University Nijmegen, is a very active researcher in the field of computer assisted language learning in general, and of Pronunciation Error Detection (PED) in particular. His keynote paper gives an overview of the algorithms, possibilities, limitations and challenges of PED for non-native or pathological speakers.

Florian Hoenig is researcher in the Speech Processing and Understanding group at the Pattern Recognition Lab of the Friedrich-Alexander University Erlangen-Nuremberg, where his work is mainly focused on the analysis of speech at the suprasegmental level. His keynote paper presents methods for and uses of automatic assessment of non-native prosody, hence broadening the focus of error detection from the segmental to the supra-segmental level.

Jack Mostow is Research professor at Carnegie Mellon University, where his primary research focus is on computer-assisted reading tutoring in the project LISTEN. His keynote paper gives a thorough overview of considerations and implementations of automatic analysis of children's read speech and feedback based on such analysis.

Lewis Johnson is Chief scientist at Alelo Inc, one of the world-leading companies focused on the use of virtual worlds to teach spoken and cultural communication skills. His keynote paper describes Alelo's work and philosophy with respect to communication and pronunciation training, and in particular the balance between the two.

Bryan Pellom is Vice President of Speech Development at Rosetta Stone, a company providing interactive solutions for language learning in more than 30 language and used by millions of individuals. His keynote paper presents the development of one of these solutions, a software practising conversational English for Korean learners, in terms of detected errors, underlying speech technology and requirements for future

development.

Gary Pelton is Vice President of Product Development at Carnegie Speech, a leading industrial developer of speech technology software for assessing and teaching languages. His keynote paper describes a survey of learner errors in English consonant clusters and discusses first language impact on the errors and the implications for practicing clusters with CAPT software.

Horacio Franco is Program Director at Speech Technology & Research Laboratory (STAR), SRI International, an independent non-profit research institute performing speech technology research for different types of clients. His keynote paper gives an overview of SRI's research on phone-level pronunciation assessment.

In addition to these keynote talks, the program and these proceedings contain regular presentations on current research topics, and one demo session, in which attendees will get hands-on experience with commercial or research software targeting CAPT.

The proceedings contain the collection of 21 papers accepted for presentation at IS ADEPT. The papers have been reviewed by the scientific committee (with two or three reviewers per paper), and resubmitted after incorporating reviewer comments and suggestions. We are grateful to all of the reviewers listed overleaf for their valuable assistance to the organizing committee and the authors.

We also acknowledge the financial support from the Swedish Research Council through the Swedish Research Links project ADEPT (Audiovisual Detection of Errors in Pronunciation Training), which is a collaborative research effort shared between KTH and the University of Cairo. The symposium would not have been possible without this support.

We hope that you will have an enjoyable symposium and that you find many opportunities to discuss your work and ideas with other delegates, not the least during the common lunches and the social program. The focused scope of the symposium and its small scale will hopefully make many new exchanges of ideas possible.

We finally hope that you will get a taste of what Stockholm has to offer its visitors in the summertime, and in particular on the Swedish National day on June 6th.

Welcome!

Olov Engwall
Chair of IS ADEPT

Organizers

- Olov Engwall, KTH, Sweden
- Christos Koniaris, KTH, Sweden
- G. Ananthakrishnan, KTH, Sweden
- Sherif Madhy Abdou, Cairo University, Egypt

Reviewers

- Fadi Biadsy, Google
- Olivier Deroo, Acapela Group
- Maxine Eskenazi, CMU
- Horacio Franco, Speech Technology & Research Laboratory SRI International
- Björn Granström, KTH (Royal Institute of Trechnology)
- Florian Hoenig, University of Erlangen
- Lewis Johnson, Alelo Inc
- Adam Lammert, University of Southern California
- Joaquim Llisterri, Universitat Autonoma de Barcelona
- Roger K. Moore, Univeristy of Sheffield
- Jack Mostow, Carnegie Mellon University
- Shri Narayanan, University of Southern California
- Bryan Pellom, Rosetta Stone
- Gary Pelton, Carnegie Speech
- Gerasimos Potamianos, NCSR Demokritos
- Helmer Strik, Radboud University, Nijmegen
- Silke Witt, Fluential Inc
- Joseph Tepperman, Rosetta Stone
- Khiet Troung, University of Twente
- Preben Wik, Artificial Solutions

Symposium program

Location: Fantum, Lindstedtsv. 24, 5th floor, KTH (labeled 1 on the map below)

Wednesday June 6

14:30-15:00 Symposium opening:
Welcoming plus presentation of keynote speakers and attendees

15:00-15:45 Keynote 1:
Silke Witt
Automatic Error Detection in Pronunciation Training: Where we are and where we need to go

16:00-16:45 Keynote 2:
Helmer Strik
ASR-based systems for language learning and therapy

17:00-18:00 Symposium reception:
KTH, main courtyard (weather permitting)

18:30- Celebration of Swedish National Day at Skansen
Optional (not part of the symposium program)

Festivities at Solliden stage in the presence of the Royal Family.

The tradition of celebrating a National Day in Sweden was born at Skansen (labeled 2 on the map below), the world's first open air museum, displaying Swedish traditions, heritage and animals. Artur Hazelius, the founder of Skansen, wanted to establish an annual day that would be a patriotic focus for the nation and he chose the 6th of June, the date of the election of King Gustav Vasa in 1523.

6th of June is celebrated in traditional fashion: His and Her Majesty the King and Queen join the procession to Skansen, where there are speeches, flag ceremony, singing, brass bands and folk music. The celebration is broadcasted live on Swedish Television.

Thursday June 7

9:00-9:45 Keynote 3:
Bryan Pellom
Rosetta Stone ReFLEX: Toward Improving English Conversational Fluency in Asia

10:00-10:45 Keynote 4:
Florian Hönig
Automatic Assessment of Non-Native Prosody - Annotation, Modelling and Evaluation

10:45-11:15 Tea time/Coffee break

11:15-12:00 Presentation session 1: Analysis methods
Christos Koniaris, Olov Engwall and Giampiero Salvi
On the Benefit of Using Auditory Modeling for Diagnostic Evaluation of Pronunciations

Amalia Zahra, Joao P. Cabral, Mark Kane and Julie Carson-Berndsen
Automatic Classification of Pronunciation Errors Using Decision Trees and Speech Recognition Technology

Keelan Evanini and Becky Huang
Automatic Detection of [θ] Pronunciation Errors for Chinese Learners of English

12:00-13:30 Lunch

13:30-14:15 Keynote 5:
Garrett Pelton
Mining pronunciation data for Consonant cluster problems

14:30-15:10 Presentation session 1: Articulation-based analysis and feedback
Tsuneo Nitta, Silasak Manosavan, Yurie Iribe, Kouichi Katsurada, Ryoko Hayashi and Chunyue Zhu
Pronunciation Training by Extracting Articulatory-Movement from Speech

Olov Engwall
Pronunciation analysis by acoustic-to-articulatory feature inversion

Sherif Abdou, Mohsen Rashwan, Kamal Jambi and Hassanin Al-Barhamtoshy
Enhancing the Confidence Measure for An Arabic Pronunciation Verfication System

15:10 Transportation to Stockholm City Hall

16:00-18:00 Visit of Stockholm City Hall
Guided visit of the interior 16:00-16:45, followed by self-guided tour of the exterior 17:00-18:00

18:00-20:30 Dinner cruise to the Royal Castle at Drottningholm
M/S Prins Carl Philip departs from Stadshusbron, next to the Stockholm City Hall, for a 2.5 hour trip on the lake Mälaren that will take us to the world heritage site of Drottningholm and back. The City Hall and the departure point for the cruise are labeled 3 on the map on the previous page.

A three-course dinner with Swedish tastes will be served during the trip:
* Gravlax (marinated salmon) with a sour cabbage salad and mint
* Archipelago fish casserole with blue mussels and prawns served with aioli
* Raspberry parfait with chocolate mousse and blueberries tossed in lime

Please signal to the organizers before the boat trip if you have dietary restrictions that are in conflict with the above menu.

Friday June 8

9:00-9:45 Keynote 6:
Lewis Johnson
Error Detection for Teaching Communicative Competence

10:00-10:45 Keynote 7:
Jack Mostow
Why and How Our Automated Reading Tutor Listens

10:45-11:15 Tea time/Coffee break

11:15-12:10 Presentation session 3: CAPT systems and users
Sherif Abdou and Mohsen Rashwan
Performance Evaluations For A Computer Aided Pronunciation Learning System

Nikos Tsourakis
A Game on Pronunciation Feedback in Facebook

Mark Kane, Zeeshan Ahmed and Julie Carson-Berndsen
Underspecification in Pronunciation Variation

Mansour Alsulaiman, Mohamed Bencherif, Ghassan Al Shatter, Saad Al-Kahtani, Ghulam Muhammad, Zulfiqar Butt and Mohamed Al-Gabri
Automatic identification of Arabic L2 Learners Origin

12:10-13:30 Lunch

13:30-15:00 Demo session
Joao Cabral, Mark Kane, Zeeshan Ahmed, Mohamed Abou-Zleikha, Eva Szekely, Amalia Zahara, Kalu Ogbureke, Peter Cahill, Julie Carson-Berndsen and Stephan Schlogl
Using the Wizard-of-Oz Framework in a Pronunciation Training System for Providing User Feedback and Instructions

Jacques Koreman, Olaf Husby and Preben Wik
Comparing sound inventories for CAPT

Thomas Hansen
The Danish Simulator - learning language and culture through gaming

Florian Hönig
Dialog of the Day

Bryan Pellom
RosettaStone ReFLEX

Lewis Johnson
Alelo language and communication training

Garett Pelton
Carnegie Speech NativeAccent

15:00-15:45 Keynote 8:
Horacio Franco
Adaptive and Discriminative Modeling for Improved Mispronunciation Detection

16:00-16:30 Symposium closing and final remarks

Contents

Automatic Error Detection in Pronunciation Training: Where we are and where we need to go

Silke M. Witt
Fluential, Inc
Sunnyvale, USA
switt@fluentialinc.com

***Abstract* — This paper discusses the state of the art of research in computer assisted pronunciation teaching as of early 2012. A discussion of all major components contributing to pronunciation assessment is presented. This is followed by a summary of existing research to date. Additionally, an overview is given on the use of this research in commercial language learning software. This is followed by a discussion of remaining challenges and possible directions of future research.**

Keywords – Pronunciation error detection, automated error detection, computer assisted language learning, Computer Assisted Pronunciation Training (CAPT)

I. INTRODUCTION

In the wake of tremendous improvements in computing power and multi-modal applications, there has also been a renewed interest in computer-assisted pronunciation teaching (CAPT) applications in recent years. With increasing globalization, there has also been a significant increase in the demand for foreign language learning, one aspect of which is pronunciation learning. Effectively teaching pronunciation typically requires one-to-one teacher student interactions, which for many students is unaffordable. For this reason, automatic pronunciation teaching has been a focus of the research community, bringing together researchers from a number of disciplines: speech recognition, linguistics, psycholinguistics, and pedagogy, as well as auditory and articulatory research.

II. A SHORT HISTORY OF CAPT

Research work on automated pronunciation error detection and pronunciation assessment started in 1990's with a flurry of activities in late 90's to early 2000, see references [1] to [8] and [9]. A detailed list of early references can also be found in [10]. Since there are a large number of publications in this area, it was only possible to quote some representative examples of papers. The author apologizes for any omissions. In the early 2000's commercialization of CAPT proved difficult and thus research activities, too, slowed down. With increased computing power, mobile devices and improved speech recognition, interest picked up again about five years ago, leading to the founding of an ISCA special interest group called SLaTE (Speech and Language Technology for Education) in 2007.

References [11], [12] and [13], provide a very thorough and in-depth overview of the work up to 2009. Since pronunciation error detection and teaching in its entirety is a difficult problem, past work has often only addressed components of this field such as phoneme level pronunciation error detection or prosodic error detection.

The next section will discuss the different components that contribute to pronunciation, followed by a discussion of many features that have been proposed to measure these pronunciation components. Then section IV presents a discussion of existing research for these various components, followed in section V by an overview of commercial systems that employ some of this research. Section VI then discusses remaining challenges in CAPT.

III. WHAT IS PRONUNCIATION?

Pronunciation is a general term that covers a number of different components and can be measured with many different features. In addition, the term 'pronunciation error' is difficult to quantify; i.e. there is no clear definition of right or wrong in pronunciation. Rather there exists an entire scale ranging from unintelligible speech to native-sounding speech.

A. Types of Pronunciation Errors

Figure 1 below illustrates all different aspects of pronunciation errors that need to be addressed in a successful training and assessment situation. Pronunciation errors can be divided into phonemic and prosodic error types.

On the phonemic side there are the 'severe' errors where phonemes might be substituted with another phoneme, deleted or inserted. Then there are the 'errors' on a smaller scale where the correct phoneme is more or less being spoken, however, the sound of it is still different enough from a native speaker's pronunciation that it is noticeable that a speaker still has an accent.

On the prosodic side a non-native accent can be categorized in terms of stress, rhythm and intonation. All such errors are closely linked which is indicated by the circles in Figure 1. This fact makes pronunciation a multi-dimensional problem that is difficult to pin down with a single approach. Rather, a successful system will require a combination of many different techniques.

One additional challenge in pronunciation error detection is that a phoneme represents the smallest possible unit

compared to the syllable, word and sentence level. The shorter the unit, the higher will be the variability in the judgment of the pronunciation quality. Even human judges have been shown to have low inter-rater and intra-rater correlations. Consequently pronunciation error detection at the phoneme level is a much harder task than measuring pronunciation fluency across multiple sentences. Kim et al. [7] showed that the error detection accuracy can be significantly improved if all realizations of the same phoneme in a student's speech sample are scored at speaker level, that is the final phoneme score is an average of all same phoneme occurrences of one speaker. They found that after about 300 instances of a given phoneme, the correlation with human ratings reaches 0.8, while the correlation of the rating for a single phoneme instance is as low as 0.5. This approach however has the disadvantage of ignoring the fact that pronunciation errors depend on surrounding phonemes and spelling-to-pronunciation rules.

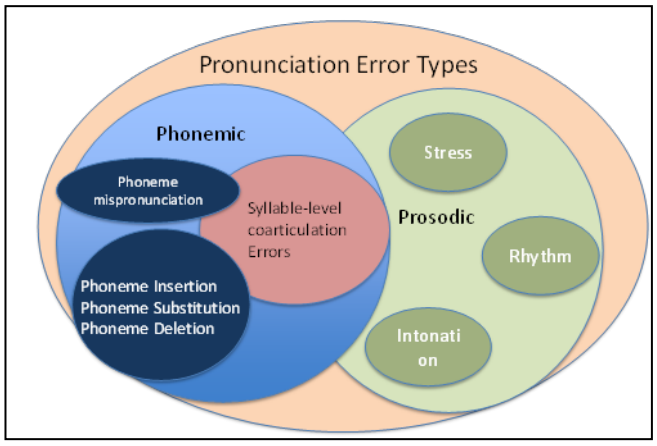


Figure 1: Types of pronunciation errors

B. *Native-like versus intelligible pronunciation*

In recent years, a discussion has started whether the goal of pronunciation teaching is to sound just like a native speaker or whether the teaching should mostly focus on the intelligibility of the student. Currently, the agreement appears to be that intelligibility is an essential component of communicative competence. While aiming to sound just like a native speaker is important, especially for more advanced students, see [14], it is clearly less critical than basic intelligibility. Raux et al. [15] explored the relationship between error rates and intelligibility and found that errors related to prosodic features, such as vowel insertion, impact intelligibility more than segmental errors, such as phoneme substitution, for example replacing 'A' with 'ER". The authors present a probabilistic model that helps to predict intelligibility based on the student's errors.

Koniaris et al. [16], used a model of the human auditory system to identify those pronunciation errors that are most noticeable to native speakers. This model provides a distance measure between non-native and native speakers which takes into account the perception of sounds by native speakers.

C. *Pronunciation features*

One possibility is to visualize the pronunciation of a unit, phoneme, syllable, word or sentence as a cluster in an N-dimensional space, where each dimension represents a pronunciation feature. The values for each pronunciation feature for native speakers will vary within a given range. Thus taking all features with their variable range together, native pronunciation can be seen as an irregularly-shaped cluster in the N-dimensional space. With this image in mind, the task of assessing non-native pronunciation can be seen as measuring the distance of the non-native speaker's pronunciation to the aforementioned cluster of native speakers. The smaller this distance, the more 'native-like' the pronunciation will be.

A large number of metrics for measuring these dimensions of pronunciation has been used over the years. An excerpt of such metrics is shown in Table 1 below. The table content is by no means complete; it is intended to demonstrate the large variety of metrics and thus the large number of pronunciation dimensions. Particularly on the prosodic side there are many similar, but slightly differently defined features that have been used.

TABLE 1: EXAMPLES OF FEATURES USED FOR PRONUNCIATION SCORING

Feature Category	Feature Name
Phonemic	Phone-level log-likelihood scores, GOP
	Vowel durations, duration trigrams
	Phoneme pair classifiers
	spectral features (formants)
	Articulatory-acoustic features
Prosodic (Intonation, Stress, fluency)	distances between stressed and unstressed syllables
	Mean, max, min power per word (energy)
	F0 contours (slope and maximum)
	rate of speech (words per second/minute)
	Trigram models to model phoneme duration in context
	Phonation/time ratio, mean phoneme duration
	Articulation Rate (phonemes/sec)
	Mean and standard deviation of long silence duration
	Silences per second
	Frequency of disfluencies (pauses, fillers etc)_
	Total and mean pause time (i.e. duration of interword pauses)

The next section will discuss in detail different metrics for the types of pronunciation errors using these features in various ways.

IV. EXISTING RESEARCH ON PRONUNCIATION ERROR DETECTION

A. *Likelihood-based scoring*

The initial work on this topic in the 1990's saw the creation of several likelihood-based phoneme-level error detection algorithms. For example, Kim et al. [7] presented three HMM-based scores: a) a HMM-based log-likelihood score, b) a HMM-based log posterior score, which later on has

become a de-facto standard, since it was shown that it had the highest correlation with human scores (this work scores the pronunciation quality of a given phoneme over many instances of the pronunciation of a given phoneme), and c) a third score based on segment duration. Similarly, the 'GOP' (goodness of pronunciation) score also uses a log-likelihood based score, [10]. Likewise Kawai et al. [17] also used log-likelihood scores in forced alignment mode. Expanded versions of likelihood-based scores were also successfully used by Mak et al. [18].

B. L1-independent approaches

One of the core decision points for pronunciation error detection is whether to build a system that is L1 (i.e. the native language) dependent or not. While it is preferable to have an L1 independent system in order to minimize the commercial implementation challenges, better performance has been found with methods that take L1 into account. In addition to likelihood-based scores, which are in most cases L1-independent, there are several additional scoring approaches that are L1-independent. Cucchiarini et al. [19] also utilized a manually annotated corpus of non-native speakers of Dutch to generate statistics on both the frequency and the context of pronunciations errors. They showed that there is a good amount of overlap between manually derived error types in the linguistic literature and such automatically derived error types. In a recent approach, Li et al. [64] combined log-likelihood scores and fluency scores, like rate of speech and trigram phone duration model, in order to score pronunciation and was able to achieve a correlation at sentence-level with human ratings of 0.84. Similarly, Cincarek et al. [20] uses a classifier-based approach that combines log-likelihood and different duration scores in order to calculate mispronunciation probabilities of phonemes across multiple utterances. Lastly, Cincarek et al. [20] also applied a L1-independent scoring mechanism based on a combination of loglikelihood and duration scores to identify common mispronunciation patterns for a given language.

C. L1 Dependency

There exists a fairly large number of work that is L1-dependent since that approach has traditionally yielded a higher accuracy than L1-independent approaches. For example, Ito et al. [21] manually derived a set of mispronunciation rules for a given L1/L2 pair and used those for clustering error rules using a decision tree. This approach resulted in increased pronunciation error detection accuracy.

IV.C.1 Automatic generation of data for L1/L2 pairs

Taking into account L1 has two main advantages: Firstly, if L1 is known, one can utilize acoustic models that are a mixture of L1 and L2 ([10], [22], [23]) and have improved speech recognition accuracy, which in turn enables recognition of less constrained utterances, which allows for greater freedom in the selection of pronunciation learning exercises, in particular for assessing fluency. Secondly, the set of common pronunciation errors tend to be typical for a given L1 and very different between different L1, i.e. a German speaker will make very different English pronunciation errors than a native speaker of Chinese or Hindi. Thus, knowledge of L1 enables to provide tailored pronunciation exercises. For example Husby et al. [24], created a tool called L1-L2*map* that contains manually entered data on likely mispronunciations for a given L1 when learning Norwegian. This data was then used to create a list of expected pronunciation errors. Likewise, Neri et al. [25] conducted a similar analysis to identify L1 specific groups of commons errors for students of Dutch.

In recent years, there have been a number of approaches to automate the process of identifying typical error patterns for a given L1-L2 pair. Lo et al. [26] and Harrison et al. [27, 28] have utilized an alignment of canonical pronunciations with manually annotated pronunciations of non-native speech to automatically generate mispronunciation rules. Such rules are then used in extended recognition networks to identify pronunciation errors. One advantage of using these recognition networks is that if an error is identified the type of error is also known and can be used for diagnosis.

Quian et al. [29] explored an alternative method to generate mispronunciation lexica. They used joint sequence multigrams to perform a grapheme to mispronunciation conversion and showed that this approach can slightly improve performance both in terms of accuracy as well as reduction in false alarm and false rejection rates. However, all these approaches still require a manually annotated corpus of non-native speech which is expensive and time-consuming to create.

Along the same lines, Stanley et al. [30] conducted research in finding mechanisms to automatically model phonological errors. The authors showed that applying statistical machine translation significantly improved the precision and recall for pronunciation errors, while the accuracy was similar to the accuracy of the extended recognition networks.

D. Classifier-based scoring

While likelihood-based pronunciation scoring has the advantage of being L1-independent and very easy to compute, it has been found that the calculated scoring is not capable of identifying the error type that has occurred.

In order to address this problem, there have been a number of studies that employ classifiers for specific phoneme pair contrasts that represent common error types. For example, Franco et al. [31] built a set of classifiers for Dutch vowel contrasts and found that adding MFCC as well as phonetic features in addition to ASR features to train classifiers gave the best classification results. Likewise, Truong et al. [32] developed a L1-independent classifier utilizing a number of acoustic-phonetic features for each expected phoneme error combination. This classifier has been shown to outperform previous approaches, but does have the drawback that common errors for a given L2 have to be known and that separate classifiers for each error type are necessary. A similar classifier for Norwegian is presented by Amdal et al. [33].

For more recent work on error detection with the help of classifiers, see Strik et al. [34]. The authors compared the

scoring accuracy for four different classifiers for a set of manually identified problem phoneme pairs for non-native speakers of Dutch. This work demonstrated that LDA based classifiers can outperform log-likelihood-based scoring.

Similarly, Yoon et al. [35] trained a landmark-based SVM classifier on an expected set of distortion errors, where a landmark is a sudden signal change such as a stop release. This approach, however, required knowledge of mispronunciation rules for a given L1 – L2 language pair.

E. Non-native acoustic modeling

If a CAPT system allows freely spoken utterances from the student, non-native acoustic modeling is required. Hui et al. [22] showed that using standard adaptation algorithms such as MAP or MLLR yields substantial recognition accuracy improvements.

Likewise, Saz et al. [23] showed that going from speaker independent to speaker dependent recognition via MAP almost reduces the phoneme recognition error rate in half. Interestingly, there was little difference if the adaptation was conducted on all available material for a given speaker or if words that had labeled mispronunciations were excluded. Such results are encouraging, because it shows that un-supervised adaptation, even if it adapts to acoustic data that includes pronunciation errors, still yields a significantly better recognition performance than no adaptation.

F. Text independence

Up to now, little work has been attempted to assess the pronunciation quality of unconstrained spontaneous speech. However, for more advanced pronunciation learning activities, it is a requirement to have students speak text freely as opposed to reading a text.

In order to do so it has been proposed to use a sequence of two different recognition tasks, see the work by Moustrofas et al. [36] and Chen et al. [37]. First, the non-native speech has to be recognized irrespective of any pronunciation errors. This is typically done with acoustic models adapted to the particular characteristics of the speaker. Secondly, the recognized text is used to perform recognition in forced-alignment mode and to calculate the pronunciation 'correctness' based on one of the many algorithms proposed for this task.

G. Prosodic pronunciation error detection and feedback

A very detailed discussion of CAPT systems that provide prosodic feedback can be found in [12].

Bernstein et al. [38] have shown that there appears to be a linear relationship between fluency measures (such as listed in Table 1) and human judgments of proficiency. Also human-ratings of fluency have been found to be reliable with inter-rater correlation above 0.9, [3]. These results show the importance of measuring fluency as part of any pronunciation assessment exercise. Recently, there has been more interest in exploring automated methods to measure prosodic features of pronunciation. For example, Levow et al. [39], used a SVM-based classifier for pitch accent recognition. Hönig et al. [65] used a large feature set based on duration, energy, pitch and pauses to detect word accents. In more recent work, Hönig et al. [66] employed a discriminative approach that uses a large number of specialized rhythm features as well as general prosodic features to create a comprehensive metric of prosodic pronunciation quality. Bonneau et al. [40] presented a system that teaches fluency with several different methods of modifying the phoneme durations and F0 contour of a learner's speech in order to demonstrate to the student (in their own voice) what their pronunciation should sound like. Initial results from a pilot study seem promising, but a larger follow-up study is needed to confirm the initial findings.

Another aspect of pronunciation not discussed so far, are tones in tonal languages like Chinese. Mixdorff et al. [41] and Hussein et al. [42] conducted initial work to detect tone errors by German learners of Mandarin Chinese. The main challenged encountered here (as in a number of other CAPT system) was a high rate of false hits.

Very recent work by Engwall [43] uses audiovisual articulatory feature inversion to estimate the learner's current articulation and to provide audiovisual feedback by showing the movement of the tongue with computer animations. An initial evaluation seemed to show that such feedback on tongue movements helped students to improve their tongue position and thus their articulation and pronunciation.

H. Corrective feedback

Corrective feedback can only be effective, if the student is also able to perceive the difference non-native and native speech, see [42, 24, 44]. For example, before being able to learn the tones of Mandarin Chinese, a student must be able to perceive the tones. Thus, a CAPT system needs to include perception training as part of corrective feedback components.

One of the earlier systems that not only attempts to detect mispronunciations but also give the student some information as to how to correct the mispronunciation, is the PLASER system, [18]. While students liked the system and 77% of the participants of a test study believed their pronunciation to have improved, robustness and correct error detection at a phoneme-level were identified as problem areas.

Bodnar et al. [45] tested the feasibility of ASR-based corrective feedback via a virtual teacher with regard to teaching L2 syntax and found to be effective within the scope of a small test study. The future goal for this system is to build it out as a platform to test a variety of different feedback strategies. Similarly, the Euronounce system by Demenko et al. [46] uses several methods of corrective feedback, but this time the feedback focusses on prosody with the help of 'pitch-line', an approximation of intonation contours that attempts to only show the relevant components of intonation (pitch-accents, boundary tones).

In order for corrective feedback around tongue positioning to work, Ouni [47] has shown that if students receive short, specific training for tongue gestures, it significantly increases the awareness of tongue positioning and thus the effectiveness of corrective feedback with the help of tongue position images/videos increases.

I. Interactive CAPT system design

When creating a system for computer-assisted pronunciation teaching, understanding the language learner's requirements and motivation are important in order to achieve any lasting success. There has been limited research on how to automatically teach pronunciation, i.e. what teaching methods or exercises are effective under various different circumstances. Derwing et al. [48] discuss the challenges in pronunciation teaching in general (i.e. independent of automation). They also called for more in-depth research around questions such as intelligibility, functional load and lasting impact of different pronunciation teaching approaches. Strik and al. [63] present a discussion of issues specific to design and pedagogy in the context of automated pronunciation teaching.

Language students have repeatedly expressed the desire to be told their pronunciation errors so that they know what to focus on. Accordingly, Neri et al. [49] showed that implementing corrective feedback even if in a limited form, does improve the pronunciation quality of students on an individual phoneme level and has a positive impact on user motivation.

The presentation layer of a CAPT system will greatly influence the acceptance rate of a system. Eskenazi et al. [5] and Yoon et al. [50] present some initial discussion of user interface design questions in conjunction with their own conclusions as to what to implement.

Sonu et al. [51] showed that both minimal pair based training and sentence level training is effective in order to improve a student's perception skills. Beyond that, there has been little work to incorporate pronunciation as part of dialog-based fluency training by engaging students in an interaction with a virtual tutor. A very early system that showed the feasibility of such an approach can be found in [52] and [53] where students had to traverse up to 7 states in order to complete an encounter. Likewise, Raux et al. [54] present an interesting approach to providing lessons within a dialog system by responding to ungrammatical sentences with a confirmation prompt that emphasizes the error. However, too high a non-native recognition error-rate prevented the effectiveness assessment of this approach.

Creating lesson material is a complex and time-consuming task. An attempt to help automating this process has been made in both [55] and [62]. Saz et al. [62] automatically identified confusable contexts that consist of an original sentence and an automatically generated sentence with a minimal pair difference. This can help students to focus on critical pronunciation errors that can cause a larger degree of misunderstanding than other pronunciation errors. There exists a large body of research on the lesson authoring for language learning, see for example Roseti et al. [56] who outline a multimedia authorship tool. Lastly, Johnson et al. [67] have built a lesson authoring system that incorporates both pedagogical considerations as well as Alelo's pronunciation teaching technology into new lessons. Alelo's products are one of the very few examples that have incorporate pronunciation learning in interactive multi-media dialog systems.

V. Existing Commercial Applications

Pronunciation error detection has two main commercial usages: (1) as part of pronunciation assessment and (2) as part of pronunciation teaching. Each application comes with a number of challenges, particularly on the pronunciation teaching side.

TABLE 2: SUMMARY OF EXISTING COMMERCIAL CAPT SYSTEMS

Product Name & Link	Company	Languages	Description
Versant and VersantPro	Pearson	English, Spanish, Arabic	Automated pronunciation assessment, measures speaking as well as listening
SpeechRater Engine (**ets.org/research/topics/as_nlp/speech)**	ETS	US English	Automated pronunciation assessment as part of standardized tests Pronunciation learning via AMEnglish.com includes training on stress, rhythm, intonation Part of TOELF since 2006
EnglishCentral	EnglishCentral	US English	English learning website, Assigns pronunciation score at sentence level, Tracks progress over time
CarnegieSpeech Assessment Climb Level 4 NativeAccent SpeakRussian SpeakFarsi	CarnegieSpeech	Russian, Farsi	Pronunciation assessment as well as pronunciation teaching. Feedback at phone and sentence level Prosody?? Measurings pausing and duration
EduSpeak	SRI StarLab	Adults: American English, Latin American Spanish, French, German, Chinese (Mandarin), Arabic (Egyptian), UK English, Australian/NZ English, Japanese, Swedish, Tagalog (Filipino) Children: American English (Ages 4 to 15)	Acoustic modeling of childrens' speech
RosettaStone Totale.	RosettaStone (rosettastone.com)	Arabic, Chinese (Mandarin), Dari,Dutch, English (US,UK), Filipino, French, German, Greek, Hebrew, Hindi, Indonesian, Irish, Italian, Japanese,Korean, Latin, Pashto, Farsi, Polish, Protuguese, Russian,Spanish (LA and Spain), Swahili, Swedish, Turkish,Urdu, Vietnamese	Immersion approach, all teaching in target language
Spexx (speexx.com)	Digital Publishing	English, Spanish, French, Italian, German	Has 12 L1 language supports. Online programs, not software package. Also uses ASR for pronunciation training. Does word-level scoring with green.yellow, red highlighting).
TellMeMore v10.0 .tellmemore.com)	Auralog	Spanish, French,German, Italian, English, Dutch, Chinese, Japanese, Arabic	Front &sideview visualization of words, audio and F0 tracking.
EyeSpeak (eyeSpeakEnglish.com)	EyeSpeak	US and British English	Audio comparison, measures each phoneme, timing, loudness. Student can listen to each phoneme segment, visual cross-section of mouth for each sound, pitch tracking
Tactical Iraqi, Dari and Pashto	Aleho (alelo.com)	Iraqui, Dari and Pashto	Pronunciation teaching and immediate corrective feedback embedded in interactive, 3D video games.

In the area of automated language skill assessment or pronunciation assessment, there has been quite some success in bridging the common gap between research and commercial development. The "phonepass" product suite from Pearson (formerly Ordinate) that is based on analyzing about 10 minutes of audio has been shown to measure pronunciation skills as reliably as human judges, see Bernstein et al. [57],[58],[59]. Additionally, ETS (Educational Testing Services) and Pearson utilize complex algorithms to measure the pronunciation quality of students based on analyzing up to 10 minutes of speech. It has been shown that such algorithms can assess the pronunciation quality of a student as reliable as a trained human expert; see for example [58].

Several commercial language learning software packages have automated pronunciation error scoring tools incorporated. For example TellMeMore (previously Auralog), has exercises that allow the student to record their utterances, they display the wavefile recording and F0 contour, so that the student can compare those to the master recording and F0 contour. This exercise also gives the student a score for the entire word, but there is no phoneme level error feedback. Also, this exercise is L1 independent. A similar approach is being used in RosettaStone as well as EyeSpeak and Speexx. EyeSpeak has an interesting feature that displays the tongue position of the student in contrast to the teacher. In summary, recording a student's wavefile and allowing the student to compare their wavefile and F0 contour in terms of stress and duration with a teacher's example is a well-established practice in commercial systems. However, the challenges lie with the reliability of the judgment scale, especially when it comes to accented speech that is close to native speech.

VI. CHALLENGES IN PRONUNCIATION ERROR DETECTION

The advantages of automated assessment are that CAPT could potentially be more reliable than human assessment, cheaper, and typically available any time and any place.

Based on the CAPT research overview in the previous section and data from an informal survey of leading researchers in this field, a list of core challenges has been:

1. Reliable phoneme-level error detection
2. Distortion error assessment
3. Text independence
4. L1 independence
5. Integrated assessment of both phonemic and prosodic pronunciation components
6. Corrective audiovisual feedback
7. Robust, interactive system design

The following paragraphs describe these seven challenges in more detail.

A. Reliable phoneme-level error detection

As observed in several papers, see for example [7], [60], the reliability of pronunciation error detection by human experts on the level of individual phonemes is fairly low. Likewise, automated error detection is not that reliable either, consequently the correlation between the two is even less. But in order to give productive feedback about the type of error that has been made, a CAPT system needs to identify the exact error within a word. On the other hand, false positives (telling a student there was an error when there was none) are potentially quite damaging for a language student.

B. Distortion error/accent detection

There has been little work on partially mispronunciations that is identifying error sources that contribute to perceived accent or on quantifying different degrees of accent.

Especially when the accented speech gets close to the target speech, the measurement uncertainty in existing algorithm is essentially larger than the actual difference between accented and target speech. For example, Mueller et al. [60] applied phoneme-level loglikelihood scoring to speech from near native language learners. It was found that the GOP score didn't correlate with human ratings. A very similar challenge has been addressed by Yan et al. [61], who used discriminative training to increase the quality of the acoustic models for similar phonemes such as 'sh-s' or 'n-l' and resulted in a small relative improvement.

C. L1-independence

As can be seen from section 4, there has been limited work that is L1-independent and yet has similar performance to methods based on knowledge of L2. On the other hand, the cost of non-native database collection and annotation is very high and does not scale. The challenge is to develop methods that derive a set of likely errors for a given student either from knowing his native language without requiring an annotated database for this L1/L2.

D. Text-independence

Conversational language training exercises requires text-independence. This can be achieved by improved adaptation and non-native acoustic modeling followed by forced-alignment pronunciation evaluation.

E. Integrated assessment

Most early work on pronunciation error detection has either focused on segmental errors or on suprasegmental features. However, especially for intermediate and advanced language learners, the majority of the errors are actually occurring with regard to the prosody. Prosody errors in particular tend to contribute to any perceived accent more so than individual phoneme mispronunciations.

F. Corrective audiovisual feedback

As can be seen from the limited availability in commercial language learning software, providing corrective feedback for pronunciation errors is a difficult task. There exist several systems that display cross-sections of the vocal tract in the attempt to show students how to move the tongue for each sound. However, the main challenge lies in making such illustrations effective so that the student knows how to implement the instruction into their own tongue movements.

G. *Robust, interactive CAPT system design*

As the review of the literature has shown there exist a large body of diverse research on almost any aspect of CAPT. The big challenge lies in combining all effective, appropriate methods in an integrated, interactive system that provides a comprehensive suite of exercises combined with trending.

VII. Conclusion: Where we need to go

This paper summarized existing research on automated pronunciation error detection as well as some of the work on automated pronunciation error correction. The remaining challenges that need to be overcome in order to be able to develop truly useful pronunciation teaching applications have also been discussed.

Altogether, many components required for such applications already exist. However, one of the largest remaining challenges is to integrate these many components into one that ideally is L1-independent, or at least easily configured for a different L1, without requiring a manually annotated non-native database. It has been shown that many different features have been used to measure the various components of pronunciations. Thus future work may lie in combining more of such features in order to achieve a larger degree of accuracy and reliability in phoneme error detection as well as the detection of subtle degrees of fluent but accented speech.

Ideally such applications would utilize an intelligent virtual tutor that takes the role of a private tutor for the student. Such a tutor would have the means of providing corrective audio-visual feedback, including illustrating the differences between a student's pronunciation and that of a reference speaker, in a manner that helps students to figure out how to reduce their accent.

Acknowledgements

Many thanks to Jared Bernstein, Catia Cucchiarini, Farzad Ehsani, Maxine Eskenazi, Horacio Franco, Go Kawai, Yoon Kim, Antoine Raux, Helmer Strik, and Klaus Zechner for sharing their knowledge and assessment of the current state-of-the-art in pronunciation error detection.

References

[1] Catia Cucchiarini, Febe de Wet, Helmer Strik, and Lou Boves. "Assessment of dutch pronunciation by means of automatic speech recognition technology", ICSLP 1998, Sydney, 1998.

[2] Catia Cucchiarini, Helmer Strik, Lou Boves. "Automatic pronunciation grading for dutch", Proc. STiLL 1998, Stockholm, 1998.

[3] Catia Cucchiarini, Helmer Strik, and Lou Bouves. "Quantitative assessment of second language learners' fluency: An automatic approach", Jor. Acous. Soc., vol. 107, 1998.

[4] Maxine Eskenazi. "Using automatic speech processing for foreign language pronunciation tutoring:some issues and a prototype", Language Learning & Technology, 2:62–67, 1999.

[5] Maxine Eskenazi, Yan Ke, Jordi Albornoz, and Katharina Probst. "The fluency pronunciation trainer: Update and user issues", Proceedings INSTiL2000, 2000.

[6] Horacio Franco, Victor Abrash, Kristin Precoda, Harry Bratt, Ramana Rao, John Butzberger, Romain Rossier, and Federico Cesari. "The SRI EduSpeak system: Recognition and pronunciation scoring for language learning", INSTiL 2000, 2000.

[7] Yoon Kim, Horacio Franco, and Leonardo Neumeyer. "Automatic pronunciation scoring of specific phone segments for language instruction", Eurospeech, Rhodes, Greece, 1997.

[8] L. Neumeyer, H. Franco, V. Digalakis, and M. Weintraub, "Automatic scoring of pronunciation quality", Speech Communication, vol 30, p. 83-93, 2000.

[9] Horacio Franco. "Combination of machine scores for automatic grading of pronunciation quality", Speech Communication, vol 30, p. 121-130, 2000.

[10] Silke M. Witt. "Use of Speech recognition in computer-assisted language learning", unpublished thesis, Cambrige Uni. Eng. Dept, 1999.

[11] Maxine Eskenazi, "An overview of spoken language technology for education", Speech Communication vol 51, p. 832-844, 2009.

[12] Rodolfo Delmonte. "Exploring Speech Technologies for Language Learning", http://www.intechopen.com/books/speech-and-language-technologies, June 2011.

[13] John Levis. "Computer technology in teaching and researching pronunciation", Annual Review of Applied Linguistics, 27:184–202, 2008.

[14] Marianne Celce-Murcia, Donna Brinton, Janet Goodwin. "Teaching Pronunciation: A reference for teachers of English to speakers of other languages," New York: Cambridge University Press, Cambridge, 1996.

[15] Antoine Raux and Tatsuya Kawahara. "Automatic intelligibility assessmentand diagnosis of critical pronunciation errors for computer-assisted pronunciation learning", ICSLP 2002, Denver, USA, 2002.

[16] Christos Koniaris, Olov Engwall. "Phoneme Level Non-native Pronunciation analysis by an Auditory Model-based Native Assessment Scheme", Interspeech 2011, Florence, Italy.

[17] Go Kawai, Keikichi Hirose. "A CALL system using speech recognition to teach the pronunciation of Japanese tokushuhaku", Proc. STiLL 1998, Marholmen, Sweden, 1998.

[18] Brian Mak, Manhung Siu, Mimi Ng, Yik-Cheung Tam, Yu-Chung Chan, Kin-Wah Chan, Ka-Yee Leung, Simon Ho, Fong-Ho Chong, Jimmy Wong, and Jacqueline Lo. "PLASER: Pronunciation learning via automatic speech recognition.", Proc. HLT-NAACL 2003 Workshop on Building Educational Applications using Natural Language Processing, pages 23–29, 2003.

[19] Catia Cucchiarini, Henk van den Heuvel, Eric Sanders, Helmer Strik. "Error selecton for ASR-based English pronunciation training in My Pronunciation Coach", Interspeech 2011, Florence, Italy, 2011.

[20] Tobias Cincarek, R. Gruhn, C. Hacker, Elmar Nöth, and S. Nakamura. "Automatic pronunciation scoring of words and sentences independent from the non-native's first language", Computer Speech & Language, 23(1):65–88, January 2009.

[21] Akinori Ito, Yen-Ling Lim, Motoyuki Suzuki, and Shozo Makino. "Pronunciation error detection for computer-assisted language learning system based on error rule clustering using a decision tree", Acoustical Science and Technology, 28(2):131–133, 2007.

[22] Hui Ye, Steve Young. "Improving the Speech Recognition Performance of Beginners in Spoken Conversational Interaction for Language Learning", Interspeech 2005, Lisboa, Portugal, 2005.

[23] Oscar Saz, Eduardo Lleida, and William Rodríguez. "Acoustic-phonetic decoding for assessment of mispronunciations in speakers with cognitive disorders", AVFA09, 2009.

[24] Olaf Husby, Åsta Øvregaard, Preben Wik, Øyvind Bech, Egil Albertsen, Sissel Nefzaoui, Eli Skarpnes, and Jacques Koreman. "Dealing with L1 background and L2 dialects in norwegian CAPT", SLaTE 2011, Venice, Italy, August 2011.

[25] Ambra Neri. "Segmental errors in dutch as a second language: How to establish priorities in CAPT", Proceedings of the InSTIL/ICALL Symposium, Venice, Italy, 2004.

[26] Wai-Kit Lo, Shuang Zhang, Helen Meng. "Automatic Derivation of Phonological Rules for Mispronunciation Detection in a Computer-Assisted Pronunciation Training System", Interspeech 2010, Makuhari, Japan, 2010.

[27] Alissa M. Harrison, Win Yiu Lau, Helen Meng, Lan Wang, "Improving mispronunciation detection and diagnosis of learners'

speech with contect-sensitive phonological rules based on language transfer," Interspeech 2008, Brisbane, Australia, 2008.

[28] Alissa M. Harrison, Wai-kit Lo, Xiao-jun Qian, Helen Meng. "Implementation of an extended recognition network for mispronunciation detection and diagnosis in computer-assisted pronunciation training", SLaTE 2009, Birmingham, England, 2009.

[29] Xiajun Qian, Helen Meng, Frank Soong, "On mispronunciation Lexicon Generation using joing-sequence Multigrams in Computer Aided Pronunciation Training (CAPT)", Interspeech 2011, Florence, Italy, 2011.

[30] Theban Stanley, Kadri Hacioglu, and Bryan Pellom. "Statistical machine translation framework for modeling phonological errors in computer assisted pronunciation training system", SLaTE 2011, Venice, Italy, August 2011.

[31] Jost van Doremalen, Catia Cucchiarini, Helmer Strik. "Automatic Detection of Vowel Pronunciation Errors Using Multiple Information Sources", ASRU 2009, Merano, Italy, 2009.

[32] Khiet Truong, Ambra Neri, Catia Cucchiarini, Helmer Strik. "Automatic pronunciation error detection: an acoustic-phonetic approach", INSTiL 2004, Venice, Italy, 2004.

[33] Ingunn Amdal, Magne Johnsen, Eivind Versvik. "Automatic evaluation of quantify contrast in non-native norwegian speech", SLaTE 2009, Birmingham, England, 2009.

[34] Helmer Strik, Khiet Truong, Febe de Wet, Catia Cucchiarini. "Comparing different approaches for automatic pronunciation error detection", Speech Communications vol 51, p. 845-852, 2009

[35] Su-Youn Yoon, Mark Hasegawa-Johnson, and Richard Sproat. "Landmark-based Automated Pronunciation Error Detection", Interspeech 2010, Makuhari, Japan, 2010.

[36] N. Moustroufas, Vassilis Digalakis. "Automatic pronunciation evaluation of foreign speakers using unknown text", Computer Speech and Language 21, p. 219-230, 2007.

[37] Lei Chen, Klaus Zechner, and Xiaoming Xi. "Improved pronunciation features for construct-driven assessment of non-native spontaneous speech". NAACL 2009, 2009.

[38] Jared Bernstein, Jian Cheng, Masanori Suzuki. "Fluency changes with general progress in L2 proficiency", Interspeech 2011, Florence, Italy, 2011.

[39] Gina-Anne Levow. "Investigating Pitch AccentRecognition in non-native speech", ACL 2009, Singapore, 2009.

[40] Anne Bonneau, Vincent Colotte. "Automatic feedback for L2 prosody learning", http://www.intechopen.com/books/speech-and-language-technologies , June 2011.

[41] Hansjörg Mixdorff, Daniel Külls, Hussein Hussein, Gong Shu, Hu Guoping, and Wei Si. "Towards a computer-aided pronunciation training system for german learners of mandarin", SLaTE 2009, Birmingham, England, 2009.

[42] Hussein Hussein, Hue San Do, Hansjörg Mixdorff, Hongwei Ding, Qianyong Gao, Guoping Hue, Si Wei, and Zhao Chao. "Mandarin tone perception and production by german learners", SLaTE 2011, Venice, Italy, 2011.

[43] Olov Engwall, "Analysis of and feedback on phonetic features in pronunciation training with a virtual teacher", Computer Assisted Language Learning, Vol 25, No. 1, p. 37-64, February 2012.

[44] Oliver Jokisch, Hongwei Ding, and Rüdiger Hoffmann. "Acoustic analysis of postvocalic /l/ in chinese learners of German in the context of an overall perception experiment", SLaTE 2011, Florence, Italy, 2011.

[45] Stephen Bodnar, Bart P. de Vries, Catia Cucchiarini, Helmer Strik, and Roeland van Hout. "Feedback in an ASR-based CALL system for L2 syntax: A feasibility study", SLaTE 2011, Venice, Italy, 2011.

[46] Grazyna Demenko, Agnieszka Wagner, Natalia Cylwik, and Oliver Jokisch. "An audiovisual feedback system for acquiring L2 pronunciation and L2 prosody", SLaTE 2009, Birmingham, England, 2009.

[47] Slim Ouni. "Tongue gestures awareness and pronunciation training," Interspeech 2011, Florence, Italy, 2011.

[48] Tracey Derwing and Murray Munro. "Second language accent and pronunciation teaching: A Research-Based approach", TESOL Quarterly, 39(3):379–398, September 2005.

[49] Ambra Neri, Catia Cucchiarini, and Helmer Strik. "ASR-based corrective feedback on pronunciation: does it really work?", Interspeech 2006, Pittsburgh, USA, 2006.

[50] Su-Youn Yoon, Lei Chen, Klaus Zechner. "Predicting Word Accuracy for the automatic speech recognition of non-native speech", Interspeech 2010, Tokyo, Japan, 2010.

[51] Mee Sonu, Keiichi Tajima, Hiroaki Kato, Yoshinori Sagisak, "Perceptual training of vowel length contrasts of Japanese by L2 listeners: Effects of an isolated word versus a word embeded in sentences," Interspeech 2011, Florence, Italy, 2011.

[52] Jared Bernstein, Amir Naini, Farzad Ehsani, "Subarashii: Encounters in Japanese Spoken Language Education", Calico Journal, 1999.

[53] Farzad Ehsani, Eva Knodt, "Speech Technology in Computer-Aided Language Learning: Strenghts and Limitations of a New CALL Paradigm", Language Learning & Technology, 2, 1: 45-60, 1998.

[54] Antoine Raux and Maxine Eskenazi. "Using Task-Oriented spoken dialogue systems for language learning: Potential, practical applications and challenges'. InSTIL 2004, 2004.

[55] Liu Liu, Jack Mostow, and Gregory Aist. "Automated generation of example contexts for helping children learn vocabulary", SLaTE 2009, Birmingham, England, 2009.

[56] Adroaldo Guimaraes Roseti, Almir dos Santos Albuquerque, Vasco Pinto da Silva Filho, Rogerio Cid Bastos. "Multimedia Authorship tool for the Teaching of Foreign Languages and distance Learning in a Multiagent Environment, in 'Multi-Agent Systems – Modeling, Control, Programming, Simulations and Applications, Dr. Faisal Alkhateeb (Ed.), ISBN: 978-953-307-174-9, InTech, 2011.

[57] Jared Bernstein, Alistarit Van Moere, Jian Cheng. "Validating automated speaking tests", Language Testing, Vol 27, p 355-377, 2010.

[58] Jared Bernstein, Jian Cheng. "Logic, Operation, and Validation of the PhonePass SET-10 Spoken English Test", Language Testing vol. 27, July 2010.

[59] Jared Bernstein, Masanori Suzuki, Jian Cheng, and U. Pado. "Evaluating diglossic aspects of an automated test of spoken modern standard arabic", SLaTE 2009, Birmingham, England, 2009.

[60] Pieter Mueller, Febe de Wet, Christa van der Walt, and Thomas Niesler. "Automatically assessing the oral proficiency of proficient L2 speakers", SLaTE 2009, Birmingham, England, 2009.

[61] Ke Yan, Shu Gong. "Pronunciation proficiency evaluation based on discriminatively refined acoustic models", I.J. Information Tech. and Comp. Science, Vol. 3, No 2. ,www.mecs-press.org, March 2011.

[62] Oscar Saz, Maxine Eskenazi. "Identifying Confusable Contexts for Automatic Generation of Activities in Second Language Pronunciation Training," SLaTE 2009, Birmingham, England, 2009.

[63] Helmer Strik, Frederik Cornillie, Jozef Colpaert, Joost van Doremalen, and Catia Cucchiarini. "Developing a CALL system for practicing oral proficiency: How to design for speech technology, pedagogy and learners" SLaTE 2009, Birmingham, England, 2009.

[64] Hongyan Li, Shen Huang, Shijin Wang, Bo Xu. "Context-dependent Duration Modelling with Backoff Strategy and Look-up Tables for Pronunciation Assessment and Mispronunciation Detection," Interspeech 2011, Florence, Italy, 2011.

[65] Florian Hönig, Anton Batliner, Karl Weilhammer, Elmar Noeth. "Islands of Failure: Employing word accent information for pronunciation quality asessment of English L2 learners", SLaTE 2009, Birmingham, England, 2009.

[66] Florian Hönig, Anton Batliner, Elmar Nöth. "Automatic Assessment of Non-Native Prosody – Annotation, Modelling and Evaluation", ISADEPT 2012, Stockholm, Sweden, June 2012.

[67] Lewis Johnson. "Error Detecton for Teaching Communicative Competence", ISADEPT 2012, Stockholm, Sweden, June 2012.

ASR-based systems for language learning and therapy

Helmer Strik
Centre for Language and Speech Technology (CLST)
Radboud University Nijmegen, the Netherlands

***Abstract* — ASR-based CALL seems to offer many possibilities for language learning and therapy. However, in both domains the speech of the users generally differs substantially from standard speech. ASR of such atypical speech is complex and challenging. Furthermore, developing successful CALL systems requires a mix of expertise. This combination of factors has led to misconceptions and pessimism on the use of speech technology in CALL. In the current paper, we provide an overview of our research in this area, which shows that speech technology can be applied in developing useful CALL systems.**

Keywords - ASR-based CALL, language learning, therapy, atypical speech

I. INTRODUCTION

Computer Assisted Language Learning is a relatively young discipline that has already produced a considerable body of research and applications, which is attested by the numerous dedicated journals, conferences, workshops, proceedings, applications and commercial products. A general observation about this impressive output is that, overall, there are relatively fewer applications and products that address language learners' production in speaking and writing. Very often CALL systems offer practice and testing in such skills indirectly, for instance by asking learners to check different sentences and indicate which one is correct. This is obviously related to the complexity of processing the learners' output, i.e. the input to the CALL system. While it is fairly easy to process input from a mouse or touch screen, e.g. clicks and drag-and-drop, or restricted text, e.g. typed with a keyboard, which usually contains short utterances to be compared to lists of possible answers, processing unrestricted nonnative text or nonnative speech is far more complex.

However, research in second language (L2) learning has indicated the importance of skill-specific output, practice and feedback [11; 23] for learning to speak and write in the L2. In other words, learners should get the chance to extensively practice speaking and writing to try to at least approximate near-native performance. Considering that the majority of CALL systems are usually employed as supplements to traditional classroom-based teaching, it seems that especially CALL systems that address speaking proficiency would offer added value with respect to teacher-fronted classes. Because of its on-line nature, speaking practice is relatively demanding in terms of teacher time: teachers need to listen to individual learners, interact with them and provide individual feedback synchronously. For writing, on the other hand, practice can take place off line, without the teacher being present and feedback can be provided asynchronously. Against this background it is understandable that researchers have been looking for ways of providing speaking practice in CALL within the limitations of the available technology [17].

In many CALL systems that address speaking proficiency learners are encouraged to speak, but their speech output is not further processed. The rationale behind this approach is that for L2 learners producing spoken output in the L2 is in itself a worthwhile activity. Although this is probably true, as is supported by research on the importance of language output [38; 9] language learners in general prefer to check whether the speech they produced was correct or not. To make this possible CALL systems have been developed in which learners are asked to imitate examples played by the system and are invited to compare their own production with the example. This latter approach can be useful to a certain extent, but self-assessment has its limitations [13], if only because learners have difficulties in perceiving certain target language contrasts [19].

For this reason researchers and commercial companies have tried to produce CALL systems that in different ways provide some form of feedback on L2 learners' speech production. Some of the earlier systems provided visual feedback in the form of intonation contours, waveforms and spectrograms, as these could be easily achieved by employing a speech analyzer. Such feedback is still used in current systems, in spite of the fact that it remains questionable whether it is useful and effective [27; 28]. An important drawback of CALL systems that do not make use of automatic speech recognition (ASR) technology is that it cannot be verified whether the learners indeed produced the intended, target utterance. In other words, if the learner says something different from the prompted utterance, the system is not able to check that and will provide feedback on the learner's production as if it was indeed the target utterance. This may affect the credibility of the system and its pedagogical value. This is a complaint often heard from users of such systems. Therefore, researchers have been exploring how speech technology could be employed to the benefit of language learning, and in particular, speaking proficiency training.

ASR-based CALL not only offers many possibilities for language learning, but also for therapy, for people with so-called communicative disabilities. However, in both domains the speech of the users generally differs substantially from standard speech. ASR of such atypical speech is complex and challenging. Furthermore, developing successful CALL systems requires a mix of expertise. This combination of factors has led to misconceptions and pessimism on the use of speech technology in CALL (see section III).

In the current paper, we focus on ASR-based CALL. In the sections below we first present a brief history (section II), then look at ASR of non-native speech (section III), present an overview of our research in this field (section IV), and we end with a discussion (section V).

In the current paper we mainly use the term ASR-based CALL, as many others do. However, it should be noted that the methods (techniques, algorithms, tools, etc.) used 'for speech' in CALL are not restricted to ASR in the narrow sense, and that ASR is used more as a term covering speech technology in general including phonetically-based methods.

II. HISTORY OF ASR-BASED CALL

ASR-based CALL received increasing attention in the late nineties. At the CALICO conference in 1996 the CALICO - EUROCALL 'Special Interest Group' (SIG) 'Computer Assisted Pronunciation Investigation Teaching And Learning' (CAPITAL) was started, which in 1999 became the CALICO - EUROCALL - ISCA SIG 'Integration of Speech Technology in (Language) Learning' (InSTIL). Furthermore, in 1998 the 'Speech Technology in Language Learning' (STiLL) workshop was organized in Marholmen (Sweden) [www.speech.kth.se/still/]. This was the starting point of a number of STiLL and InSTiL related activities.

In 1999 a special issue of CALICO appeared, entitled 'Tutors that Listen', which focused on ASR [21]. It concerned mainly so-called 'discrete ASR', i.e. the recognition of individual words that are uttered with pauses between the words. Obviously, this is not the preferred way of communicating when learning a language. Therefore attention shifted towards continuous speech. InSTIL organized an 18 poster exhibition called 'An Illustrated History of Speech Technology in Language Learning', which was shown at EUROCALL 2001 in Nijmegen (Netherlands) and EUROSPEECH 2001 in Aalborg (Denmark). For more information on the history in this field see e.g. Delcloque [10] and Eskenazi [18].

At the Interspeech 2006 conference of the International Speech Communication Association (ISCA) in Pittsburgh there was a special session on 'Speech and Language in Education'. This was the starting point of the ISCA SIG on 'Speech and Language Technology in Education' (SLaTE) [www.sigslate.org]. SLaTE has organized several workshops since then. In addition, at the 'Innovative Use of NLP for Building Educational Applications' (BEA) workshops of the 'Association for Computational Linguistics' (ACL), now (i.e. 2012) in its seventh edition, the role of speech technology has gradually increased [www.cs.rochester.edu/~tetreaul/ academic.html]. The workshops and other activities of the SIGs mentioned above (e.g. conference special seesions), have led to many publications, see e.g. the proceedings of these events.

Speech recognition technology also gradually found its way into commercial CALL systems by companies. Well-known are 'Tell me More' by Auralog [www.tellmemore.com], 'Rosetta Stone' [www.rosettastone.com], and 'IntelliSpeech' by 'digital publishing' [www.digitalpublishing.de].

III. ASR OF NON-NATIVE SPEECH

As the quality of speech technology improved, more and more researchers tried to apply it to language learning, sometimes with disappointing results. Some researchers were skeptical about the usefulness and effectiveness of ASR-based CALL programs: evidence gathered in different lines of research seemed to confirm that either speech technology was not mature enough, or ASR-based CALL programs were not effective in improving second language (L2) skills [e.g., 3; 12]. For the sake of our own research, we studied this literature thoroughly and gradually acquired the impression that, while it is undeniable that speech technology still presents a number of limitations, especially when applied to non-native speech, part of this pessimism is in fact due to misconceptions about this technology and CALL in general.

For instance, in some studies unsatisfactory results were obtained when standard dictation systems were used for CALL [3; 12]. But such dictation systems are not suitable for L2 training, as CALL requires dedicated speech technology. Apart from the fact that the majority of dictation packages are developed for native speakers, the major problem in using this technology for CALL has to do with the different goals of dictation and CALL which require different approaches in ASR. The aim of a dictation package is to convert an acoustic signal into a string of words and not to identify L2 errors, which requires a different, more complex procedure. Consequently, the negative conclusions related to the use of dictation packages should be related to those specific cases and not to ASR technology in general.

ASR of native speech is already complex because of many well-known problems such as background sounds, (low) signal-to-noise ratio (SNR), end-point detection, pronunciation variation, and disfluencies. However, ASR of atypical speech is even more complex, since the grammar, the words used, and the pronunciation can deviate considerably, thus affecting all three 'knowledge sources' of the ASR system (language model , lexicon, and acoustic models, resp.). In the ASR community, it has long been known that the differences between native and non-native speech are so extensive as to degrade ASR performance considerably [14; 16; 22; 26; 39; 40]. Furthermore, native and non-native speech can differ in many (sometimes unexpected) ways, e.g. for non-native we found more broken words for (cold) reading, and many more filled pauses in spontaneous speech [4].

IV. ASR-BASED CALL RESEARCH

The current section presents a brief overview of our research in this field to give an idea of what can be achieved with current technology. In 1997, we started the project 'Automatic testing of oral proficiency' [41] in which we aimed at developing a system for automatic assessment of foreign speakers' oral proficiency in Dutch by using ASR technology. The results showed that automatic testing of certain aspects of oral proficiency was feasible: the scores obtained by means of ASR technology were strongly correlated with human judgments of oral proficiency. Especially oral fluency appeared to be easily predictable based on automatically calculated temporal measures [7; 8].

Pronunciation grading, as in the ATOP project, is used to calculate a score at a rather global level (e.g for a couple of utterances), which might be sufficient for testing purposes, but in general it is not detailed enough for training purposes. For training error detection is required, that is the procedure by which a score at a local (e.g. phoneme) level is calculated. For grading more global measures can be used, such as temporal measures [7; 8]. In general, the relation between human and automatic grading improves if longer stretches of speech are

used, i.e. complete utterances or a couple of utterances [see e.g., 20]. Such cumulative measures can also be adopted for error detection, for instance by combining the scores of several utterances. This can be useful to assess the problems of a specific speaker, to obtain an overview and suggest remedial exercises for the problematic cases. However, for remedial exercises immediate feedback based on local calculations is to be preferred. For pronunciation error detection, some approaches can be used: (1) focus on frequent errors, (2) use ASR-based metrics or (3) acoustic phonetic classifiers.

In the first approach, errors frequently made by language learners are explicitly taken into account [25]. For instance, in DL2, if the sound /h/ is often deleted (e.g. 'elmer' instead of 'helmer'), /g/ is often pronounced as /k/, and long and short vowels are interchanged, then these frequent errors can be included in the pronunciation models. The ASR then has to find the best path in these pronunciation networks, and can thus determine whether a pronunciation error was made.

In the second approach, ASR-based metrics are used, such as posterior probabilities and (log) likelihood ratios [20; 22; 26]. Research has shown that these confidence measures can be used for detecting pronunciation errors [20; 22; 26; 40]. A special case concerns the so-called goodness of pronunciation (GOP) algorithm [40], which has been used in several studies. We have conducted detailed studies of the GOP algorithm [15; 24; 36; 37]. If properly trained, the GOP algorithm works satisfactorily; e.g. in Dutch-CAPT system (see below) 80-95% of the sounds were classified correctly. However, there are large variations between individuals and sounds. If specific settings (thresholds) could be used for each person sound combination, better results could be achieved [24]; but in practice this is not possible. And since the GOP algorithm has some other limitations, we have been studying possible alternative measures [see e.g., 15].

The third approach, based on acoustic phonetic classifiers, is not often used in CALL applications; still it can be useful [36; 37]. We compared the results of acoustic phonetic classifiers to those obtained with the GOP algorithm, and found that results for acoustic phonetic classifiers were generally better [36; 37]. As can be expected, a combination of approaches probably yields the best results. Therefore, the challenge here is to find the proper combination of approaches and settings to achieve the best results.

Most approaches, such as the often applied (supervised) machine learning approach, require large amounts of annotated data in order to train the classifiers. Since obtaining annotated data is laborious, we have been studying other ways to carry out pronunciation detection. The acoustic-phonetic approach mentioned above is already a first step in that direction. Another approach we studied, is to use artificial errors [24]. We first obtained an overview of frequently made errors, then artificially introduced these errors into native training material, used this material to train error detectors, which were subsequently employed in the Dutch-CAPT system. Language learners then used the Dutch-CAPT system, their interactions were recorded and annotated afterwards. Analysis of these annotations showed that the performance of these error detectors in real use was comparable to the performance during development. This is remarkable, since with speech technology performance during real use is often lower than during development, and this is especially the case when there is a training-test mismatch, which was the case here (training: artificial errors in native speech, test: real errors in non-native speech). Probably, this is because we carefully introduced artificial errors that were based on analyses of actually occurring errors [24].

In the 'Dutch Computer-Assisted Pronunciation Training' (Dutch-CAPT) project [42] a pronunciation training program was developed to provide automatic feedback on segmental (phoneme) errors (see Figure 1). We evaluated this system by comparing production data by an experimental group that used the Dutch-CAPT system, with those of a control group that did similar exercises but did not get feedback on pronunciation errors. The learners in the two groups had been living in the Netherlands and had followed DL2 lessons. Already after two short sessions of about 30-60 minutes, we could observe that the decrease in the number of pronunciation errors was substantially larger for the experimental group compared to the control group that did not receive feedback) [6; 30].

Before developing a CALL system, we generally try to obtain an overview of frequent errors made by language learners by combining information found in the literature, expertise of language teachers, and analysis of data. Even if the artificial error procedure described above is used, such an overview is essential to carefully introduce the right errors in the right way. We have already derived overviews of frequent segmental errors for different combinations of first (L1) and target (L2) languages: many L1s - Dutch [27; 29], Spanish - Dutch [2], Dutch - English [5]; and also for grammatical errors in DL2 [34; 35], and segmental errors in dysarthric speech (see PEDDS project below). Deriving information on segmental errors from data was achieved through well-known procedures, while to derive information on grammatical errors from data we developed a novel procedure [34; 35].

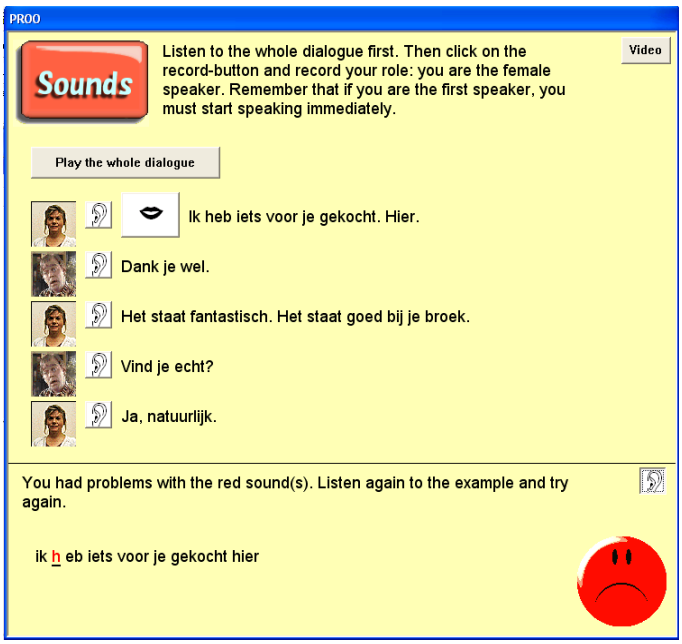


Figure 1. Screenshot of the Dutch-CAPT system. The user first watches a video, then plays a role, and gets feedback on pronunciation errors.

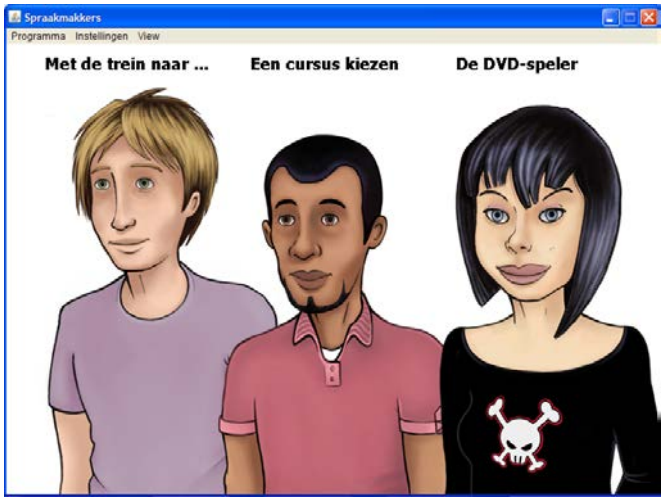


Figure 2. A screenshot of the DISCO system. The user can choose an interlocutor (‘spraakmakker’ – ‘speech buddy’) to speak to. The topics vary: a train journey, choosing a course, and going to the shop with a broken DVD player, respectively.

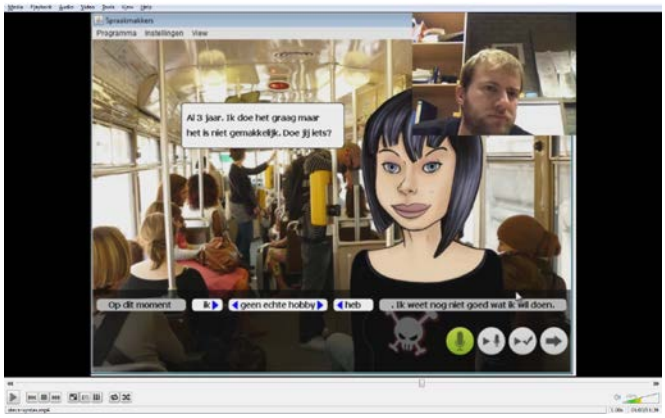


Figure 3. A screenshot of the DISCO system with a user in the upper-right corner. It concerns a syntax exercise: the user has to speak the words in the correct order.

Since good results were obtained with the Dutch-CAPT system on pronunciation, we decided to go develop a system for training not only pronunciation, but also grammar (morphology and syntax) in spoken language. To this end, we employed the overviews of pronunciation and grammatical errors mentioned above. This work was carried out in the 'Development and Integration of Speech technology into Courseware for language learning' (DISCO) project (see Figures 2 and 3), which is now almost finished [32; 33; 43]. The first user tests are encouraging, students are very positive about the system and additional evaluations will be performed soon.

In our research we develop CALL systems. In turn, we also use these CALL systems to carry out research, and the results of this research can in turn be used to improve CALL systems, and the way they are employed. We thus hope to create an upward spiral. For instance, an important issue in CALL systems for training oral proficiency is how to provide feedback. This issue is studied in the project 'Feedback and the Acquisition of Syntax in Oral Proficiency' (FASOP) [44], in which a modified version of the DISCO system is employed to conduct experiments on oral syntax practice and acquisition (see Figure 4). Dutch L2 learners are pre-tested before undergoing specific training in L2 syntax through different versions of the CALL system that provide different forms of feedback. Post-tests are then administered to determine the effects of the feedback (see Figure 5). The first results are encouraging [1].

Besides research on ASR-based CALL systems for DL2, we recently started a project on English pronunciation training for Dutch learners ('My Pronunciation Coach', MPC) [45], and the 'Lifelong Learning Programme' (LLP) project 'Games Online for Basic Language learning' (GOBL) [46] in which mini-games for language learning will be developed.

Figure 4. A screenshot of the FASOP system. Learners first watch a video clip and then answer questions. In this example, the tutor is asking ‘What does it say on the box that Melvin has packed his things in?’. To answer, learners compose an utterance using the prompt and word groups presented on the screen. All (’Allemaal’) the word groups in the blue box have to be used, and only one (’Eentje’) from the box in green.

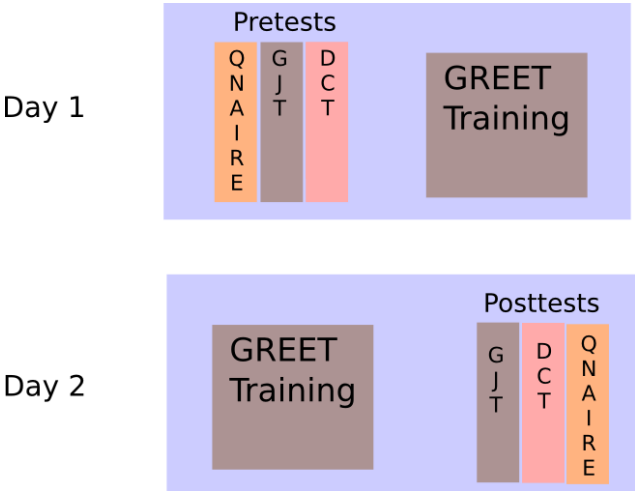


Figure 5. Overview of the FASOP experiment: ‘QNAIRE’ – questionnaire, GJT – grammatical judgment test, DCT – discourse completion test.

As explained above, non-native speech deviates from native speech in different respects. Another type of atypical speech is that produced by people with communicative disabilities (a 'speech handicap'). Similar techniques as those used for language learning can be applied in this clinical setting. For instance, in a pilot study we studied ASR of dysarthric speech. Dysarthria exists in different forms and can vary from mild to severe. If one is not familiar with the specific kind of dysarthric speech, it is usually difficult to understand the speaker in question. In our pilot study it was shown that also for dysarthric speech the performance of ASR degrades, but can be substantially improved by optimizing the ASR system for dysarthric speech [31]. The challenge here is to capture the patterns for this type of atypical speech in the models of the ASR system. This becomes more problematic if the speech (and its patterns) is not constant, e.g. in the case of progressive dysarthria. In any case, it is advisable (esp. in clinical applications, but probably also in CALL applications) to regularly update the models of the ASR system.

In the 'Pronunciation Error Detection for Dysarthric Speech' (PEDDS) project (see Figure 6), we developed technology for detecting pronunciation errors in dysarthric speech [47]. We also made a video demo to show what the possibilities are of using such technology for pronunciation training [47]. In this demo the user first watches a video (in this case an old news broadcast), then produces some utterances, gets immediate feedback on the pronunciation errors made, optionally can listen to the example utterances, and can try to pronounce the utterances again.

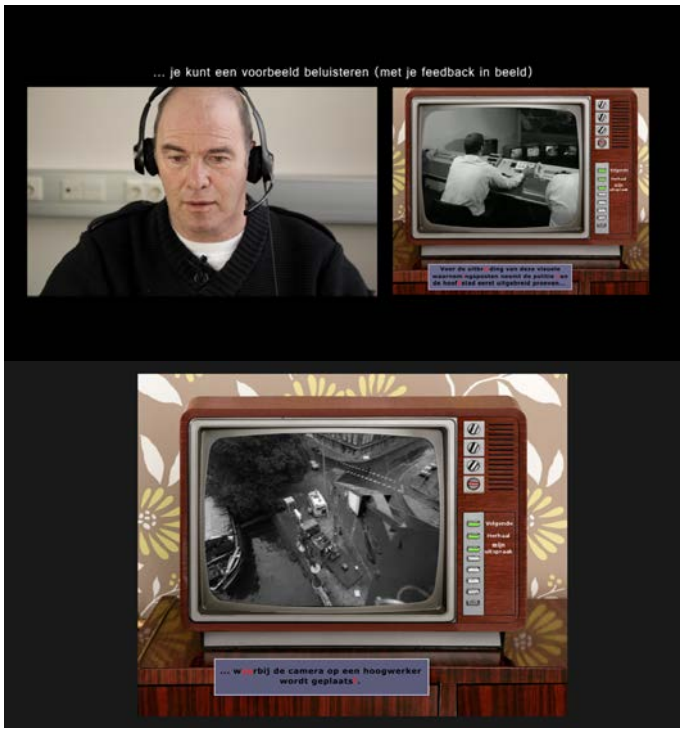


Figure 6. Two sceenshots of the PEDDS system. The user speaks utterances and gets immediate feedback on errors in its dysarthric speech.

Finally, in the 'Communication & Revalidation DigiPoli' (ComPoli) project we are developing technology to assist users in communicating with e-Health websites [48]. Nowadays, more and more people have to use websites (so called 'digipolies'), to look for information, communicate with other patients and/or experts, make appointments, etc. However, for people with communicative disabilities, this can cause problems. We will use different technologies (such as ASR, text-to-speech synthesis, and word prediction) to enhance their possibilities of communication with these websites. A first version of the website is finished, and soon we will start user trials.

V. Discussion

Above we already mentioned some reasons why developing high quality applications for atypical speech is complex, and therefore challenging; some additional issues are briefly discussed here.

To develop sound ASR-based CALL systems a mix of expertise is needed, expertise on technology for atypical speech, but also on, e.g., language acquisition, language learning, pedagogy, language course and software design, when it concerns foreign or second language learning; and similar expertise for clinical applications. In such applications eliciting speech is also challenging. It should be done in a way that is does not feel unnatural, is motivating, and, of course, effective. However, since automatic handling of spontaneous speech is not feasible yet, it should also be constrained, and the technology should be optimized for the (constrained) target speech in such a way that the system works properly. The challenge here is to develop the appropriate algorithms, and optimize them while finding the right balance between all these, often conflicting, preconditions.

Evaluation can be carried out in different ways. It is possible to evaluate the individual system components, off-line, using suitable data (speech corpora). This is generally done during development of the system. A problem is that often large amounts of suitable training material are not available. This is especially the case for detection of less frequent errors. If the interactions of the users with the system are recorded, and annotated afterwards, the same system components can also be evaluated in a more realistic context, i.e. during real use. Another possibility is to ask the system users to fill in questionnaires. More challenging is to evaluate whether the system is effective, e.g. by comparing the results of pre- and post-tests. This is what we have already done in the Dutch-CAPT, DISCO and FASOP projects (see above), in which we were able to show that such systems can be effective for pronunciation and grammar training.

Therefore, although developing CALL systems for atypical speech is complex and challenging, the overview of the projects presented above, and the positive results we obtained in these projects, makes it clear that with current state-of-the-art technology it is possible to develop useful applications for language learners (for testing and training), and persons with communicative disabilities (for diagnosis, therapy, monitoring, and AAC: augmentative and alternative communication). Such research is interesting from a scientific point of view, but obviously the resulting technology and CALL systems can be very useful for the target groups. Valorization of research and

transfer of knowledge from academia to industry are becoming more and more important, and the topics described above offer numerous opportunities in this direction.

REFERENCES

References are listed in alphabetical order, URLs at the end.

[1] S. Bodnar, B. Penning de Vries, C. Cucchiarini, H. Strik, R. Van Hout (2011) Feedback in an ASR-based CALL system for L2 syntax: A feasibility study. Proc. of the SLaTE-2011 workshop, Venice, Italy.

[2] J.M. Burgos Guillén, C. Cucchiarini, R. van Hout, H. Strik (2012) Spanish Learners of Dutch: Directions to Improve Phonology Acquisition. Submitted to Interspeech 2012, Portland, USA.

[3] D. Coniam (1999) Voice recognition software accuracy with second language speakers of English. System, 27, 49-64.

[4] C. Cucchiarini, J. van Doremalen, H. Strik (2010) Fluency in non-native read and spontaneous speech. Proc. of DiSS-LPSS Joint Workshop 2010, Tokyo, Japan.

[5] C. Cucchiarini, H. van den Heuvel, E. Sanders, H. Strik (2011) Error selection for ASR-based English pronunciation training in 'My Pronunciation Coach'. Proceedings of Interspeech 2011, Florence, Italy.

[6] C. Cucchiarini, Neri, A., Strik, H. (2009) Oral Proficiency Training in Dutch L2: the Contribution of ASR-based Corrective Feedback. Speech Communication, Volume 51, Issue 10, October 2009, Pages 853-863.

[7] C. Cucchiarini, H. Strik, L. Boves (2000) Different aspects of expert pronunciation quality ratings and their relation to scores produced by speech recognition algorithms Speech Communication 30 (2-3), pp. 109-119.

[8] C. Cucchiarini, Strik, H., & Boves, L. (2002) Quantitative assessment of second language learners' fluency: Comparisons between read and spontaneous speech. Journal of the Acoustical Society of America, 111(6), 2862-2873.

[9] K. De Bot (1996) The psycholinguistics of the Output Hypothesis, Language Learning 46, 529-555.

[10] P. Delcloque (2000), History of CALL, see www.ict4lt.org/en/History_of_CALL.pdf, and www.eurocall-languages.org/resources/history_of_call.pdf

[11] R. M. DeKeyser, Sokalski, K. J. (1996). The differential role of comprehension and production practice. Language Learning, 46 (4):613–642 (29).

[12] T. M.Derwing, Munro, M. J., & Carbonaro, M. (2000). Does popular speech recognition software work with ESL speech? TESOL Quarterly, 34, 592-603.

[13] A. Dlaska and C. Krekeler, "Self-assessment of pronunciation," System, 36, pp. 506-516, 2008.

[14] J. van Doremalen, C. Cucchiarini, H. Strik (2010) Optimizing automatic speech recognition for low-proficient non-native speakers. EURASIP Journal on Audio, Speech, and Music Processing, volume 2010 (2010), Article ID 973954, 13 pages.

[15] J. van Doremalen, C. Cucchiarini, H. Strik (2010) Using Non-Native Error Patterns to Improve Pronunciation Verification. Proc. of Interspeech 2010, Tokyo, Japan.

[16] J. Van Doremalen, C. Cucchiarini, H. Strik (2011) Speech Technology in CALL: The Essential Role of Adaptation. Interdisciplinary approaches to adaptive learning; Communications in Computer and Information Science series, 2011, Volume 26, pp. 56-69.

[17] F. Ehsani, & Knodt, E. (1998) Speech technology in computer-aided learning: Strengths and limitations of a new CALL paradigm. Language Learning & Technology, 2, 45-60.

[18] M. Eskenazi (2009) An overview of Spoken Language Technology for Education. Speech Communication, 2009.

[19] J. Flege, "Second-language speech learning: Findings and problems," In Speech Perception and Linguistic Experience: Theoretical and Methodological Issues in Cross-Language Speech Research, Winifred Strange (ed.), Timonium, MD: York Press Inc, pp. 233–273, 1995.

[20] H. Franco, Neumeyer, L., Digalakis, V., & Ronen, O. (2000) Combination of machine scores for automatic grading of pronunciation quality. Speech Communication, 30, 121-130.

[21] V. M. Holland (Ed.). (1999). Tutors that listen: Speech recognition for language learning [Special issue]. CALICO Journal, 16(3).

[22] ISLE 1.4 (1999) Pronunciation training: Requirements and solutions, ISLE Deliverable 1.4. Retrieved February 27, 2002, from http://nats-www.informatik.uni-hamburg.de/~isle/public/D14/D14.html.

[23] S. Izumi (2002). Output, input enhancement, and the noticing hypothesis. Studies in Second Language Acquisition, 24:541 – 577 (36).

[24] S. Kanters, C. Cucchiarini, H. Strik (2009) The Goodness of Pronunciation Algorithm: a Detailed Performance Study. Proc. of the SLaTE-2009 workshop, Warwickshire, England.

[25] G. Kawai, Hirose, K. (1998) A method for measuring the intelligibility and nonnativeness of phone quality in foreign language pronunciation training, Proceedings of ICSLP, Sydney, Australia, 1823-1826.

[26] W. Menzel, Herron, D., Bonaventura, P., & Morton, R. (2000) Automatic detection and correction of non-native English pronunciations, Proceedings of InSTiL, Dundee, Scotland, 49-56.

[27] A. Neri. (2007) The pedagogical effectiveness of ASR-based computer assisted pronunciation training. PhD thesis, University Nijmegen.

[28] A. Neri, C. Cucchiarini, H. Strik (2002) Feedback in computer assisted pronunciation training: Technology push or demand pull? Proc. of ICSLP-2002, Denver, USA, pp. 1209-1212.

[29] A. Neri, C. Cucchiarini & H. Strik (2006) Selecting segmental errors in L2 Dutch for optimal pronunciation training. IRAL - International Review of Applied Linguistics in Language Teaching, 44, pp. 357-404.

[30] A. Neri, C. Cucchiarini, H. Strik (2008) The effectiveness of computer-based speech corrective feedback for improving segmental quality in L2 Dutch. ReCALL, Volume 20, Issue 02, May 2008, pp. 225-243.

[31] E. Sanders, M. Ruiter, L. Beijer, H. Strik (2002) Automatic recognition of Dutch dysarthric speech: A pilot study Proc. of ICSLP-2002, September 16-20, 2002, Denver, USA, pp. 661-664.

[32] H. Strik, J. Colpaert, J. van Doremalen, C. Cucchiarini (2012) The DISCO ASR-based CALL system: practicing L2 oral skills and beyond. Proc. of the Conference on International Language Resources and Evaluation (LREC 2012)

[33] H. Strik, F. Cornillie, J. Colpaert, J. van Doremalen, C. Cucchiarini (2009) Developing a CALL System for Practicing Oral Proficiency: How to Design for Speech Technology, Pedagogy and Learners. Proceedings of the SLaTE-2009 workshop, Warwickshire, England.

[34] H. Strik, J. van Doremalen, J. van de Loo, C. Cucchiarini (2011) Improving ASR processing of ungrammatical utterances through grammatical error modeling. Proceedings of the SLaTE-2011 workshop, Venice, Italy.

[35] H. Strik, J. van de Loo, J. van Doremalen, & C. Cucchiarini (2010) Practicing Syntax in Spoken Interaction: Automatic Detection of Syntactic Errors in Non-Native Utterances. Proc. of the L2WS, SLaTE-2010 workshop, Tokyo, Japan.

[36] H. Strik, K. Truong, F. de Wet, C. Cucchiarini (2007). Comparing classifiers for pronunciation error detection. Proceedings of Interspeech 2007, Antwerp.

[37] H. Strik, K. Truong, F. de Wet, C. Cucchiarini (2009) Comparing different approaches for automatic pronunciation error detection. Speech Communication, Volume 51, Issue 10, October 2009, Pages 845-852.

[38] M. Swain (1985) Communicative competence: some roles of comprehensible input and comprehensible output in its development, in Gass, M.A., Madden, C.G. (eds.) Input in Second Language Acquisition, Rowley MA: Newbury House, 235-253.

[39] D. Van Compernolle (2001) Recognizing speech of goats, wolves, sheep and ... non-natives. Speech Communication, 35, 71-79.

[40] S. Witt (1999) Use of Speech Recognition in Computer Assisted Language Learning. PhD thesis, University of Cambridge.

[41] lands.let.ru.nl/~strik/research/ATOP.html

[42] lands.let.ru.nl/~strik/research/Dutch-CAPT/

[43] lands.let.ru.nl/~strik/research/DISCO/

[44] lands.let.ru.nl/~strik/research/FASOP.html

[45] lands.let.ru.nl/~strik/research/MPC-STW-VG2.html

[46] lands.let.ru.nl/~strik/research/GOBL

[47] lands.let.ru.nl/~strik/research/PEDDS

[48] lands.let.ru.nl/~strik/research/ComPoli

Rosetta Stone ReFLEX: Toward Improving English Conversational Fluency in Asia

Bryan Pellom
Rosetta Stone Labs
Boulder, Colorado, USA
bpellom@rosettastone.com

***Abstract*—Despite considerable spend in terms of time and money, Korean and Japanese language learners struggle to communicate effectively in English. There are several possible factors for this lack of success, ranging from overemphasis of existing curricula on nonspeaking skills to cultural factors that hinder the learning process. This paper describes Rosetta Stone ReFLEX, a novel online solution specifically designed to address the shortcomings of more traditional learning methods and improve conversational fluency. Unlike traditional methods, Rosetta Stone ReFLEX engages learners in an adaptive, 30-minute daily program that combines games and other activities that practice sound skills, simulated conversational narratives that rely on speech recognition, and one-on-one live human interaction. This paper describes the typical pronunciation error patterns made by Korean learners of English, summarizes the Rosetta Stone ReFLEX solution, and describes several of the underlying speech technologies associated with delivering the online solution. Finally, practical areas for future research as well as existing challenges are discussed.**

Keywords-speech recognition; pronunciation error detection; adaptive instruction

I. INTRODUCTION

Every year Koreans spend a tremendous amount of time and billions of dollars in pursuit of learning English. By one estimate, Koreans spent $15.3 billion in 2006 in private English tutoring, and language learning overall accounted for approximately 1.9% of the country's gross domestic product [1]. The average Korean spends over 15,000 hours learning English from middle school through college, and despite this incredible effort, a survey by Seoul's city government found that 74.2% of respondents couldn't comfortably communicate in English with foreigners [2]. Why is it that English language-learning success has been so limited in Korea? One possibility relates to motivation. Korean learners of English (KLEs) have been motivated largely by extrinsic factors, such as learning English to pass a test. English exams in Korea are used heavily as a requirement to enter college or to receive a pay raise while on the job [3]. In fact, according to [1], over 100,000 Koreans took the TOEFL (Test of English as a Foreign Language) from 2004 through 2005, accounting for 18.5% of all people who took the test worldwide. Another reason for the lack of communicative success in Korea relates to an overemphasis in the existing curricula on listening, reading, and grammar skills rather than on speaking and communication strategies [4].
Other possible factors include the lack of teachers with proficiency in spoken English, as well as large class sizes; each of which degrades the student's ability to interact with a fluent speaker and actively learn from speaking exercises.

The situation in Korea lends itself to a fairly specialized population of learners. Most adults who have studied English have mastered grammar, reading, and vocabulary skills (intermediate to advanced capability) but lack strong, spoken conversational skills (beginner to intermediate capability). One key barrier to success relates to the lack of ability to perceive and produce particular speech sounds that aren't present or phonologically distinct in the Korean language (e.g., distinguishing /ɹ/ vs. /l/). In addition, learners are simply unable to produce the natural stress, rhythm, and intonation patterns of English. Another barrier appears to be translation. Because many KLEs have not built strong spoken-communication strategies, they often revert to actively translating from their native language to English before speaking aloud. This gets in the way of generating natural and fluent conversations. Some learners are also intent on producing "perfect" English and feel uncomfortable making spoken errors in front of a native speaker. The lack of confidence by KLEs sometimes results in no response and lost opportunities to learn from spoken errors when conversing with a native speaker.

Rosetta Stone has delivered a unique, technology-enabled language-learning approach for more than 20 years. Our pedagogy is based on our belief that learning is most successful and natural when students learn in context and make direct connections between words and their meaning. As a result, our learning method works without the need for translation and focuses on implicit grammar and syntax instruction—similar to the way a child learns a language for the first time. That is, by directly associating images, text, and sounds in the target language in a meaningful context, learners can limit interference from their own native language. We call this methodology Dynamic Immersion. Our current general language-learning solution is known as Rosetta Stone Version 4 TOTALe. Available in 24 languages, this product combines several key components that we felt were useful for KLEs. First, the product includes an intelligently sequenced course that includes reading, listening, writing, and speaking activities. Throughout the course learners actively use speech recognition technology to practice speaking and to receive immediate pronunciation feedback. In TOTALe, learners participate in "listen and repeat" exercises but also produce novel phrases using knowledge acquired during the course. Finally, and perhaps most importantly, as learners progress through the curriculum, they have the opportunity to periodically schedule a live, 50-minute practice session with a tutor who is a native

speaker of the target language. This live-instruction component makes use of video-based streaming technology and connects each tutor with up to four learners who have completed similar parts of the core curriculum. The TOTALe product additionally includes a wide range of online games and activities that can be played with other language learners (or native speakers) to reinforce concepts taught in the main course. TOTALe Companion, featuring speech recognition on mobile devices, and Audio Companion, featuring TOTALe MP3 files, are provided to help learners continue to achieve success while away from their personal computers or on the go.

While TOTALe has been widely successful as an integrated and comprehensive learning solution, the needs of many Asian learners of English, including KLEs, motivated us to design an entirely new platform and curriculum experience. Unlike Version 4 TOTALe which serves many learners with differing L1 backgrounds, we desired to focus on improving the abilities of Asian learners of English to perceive and produce difficult speech sounds—those segmental patterns that may not exist in their native language and therefore are difficult for them to perceive or produce in English. We also aimed at developing a curriculum that was focused on improving the overall confidence of the learner. In doing so, we created a unique balance between providing machine-based dialogue practice using automatic speech recognition while providing one-on-one feedback using live interaction with a tutor. The overall combination of changes led us to develop a notion of a daily training session in which learners sign in for roughly 30-minute sessions that include a focus on sound skills, simulated conversational dialogue, and a live conversation with an American tutor at the end of each session.

The following sections summarize the phonetic language differences between Korean and English and common pronunciation error patterns. Next, the Rosetta Stone ReFLEX language-learning system is described and several component speech technologies used within the service are discussed in detail. Finally, active areas of future research are noted.

II. Common Pronunciation Error Patterns of Korean Learners of English

A summary of Korean and English language differences as well as typical pronunciation error patterns can be found in [6] and a detailed summary of grammatical issues can be found in [7]. The Korean language differs significantly from English along several dimensions. Korean consists of approximately 40 sounds (8 vowels, 13 diphthongs, and 19 consonants). Vowels in Korean do not appear in the word-initial position, and the language makes no distinction between long (tense) and short (lax) vowels. This aspect makes it particularly difficult for Korean learners to produce and perceive the subtle differences between the sounds /i/ and /ɪ/ for example. Diphthongs in Korean are produced by a glide (/j/ or /w/) followed by a vowel. English, on the other hand, produces diphthongs using the opposite sequence. Koreans therefore tend to perceive and produce English diphthongs as two independent syllables [8].

Unlike English, the Korean language also makes no distinction between voiced and unvoiced consonants and lacks the fricatives /f/ and /v/. Learners often substitute /p/ and /b/ instead [6]. The production and perception of the sounds /l/ and /ɹ/ are also difficult since the Korean language contains a single character that represents both /l/ and an alveolar flap, which occur only in allophonic variation in the language. KLEs also tend to substitute /l/ for /ɹ/ in word-initial positions. Other common reported errors include the substitution of aspirated /t/ for /θ/ and unaspirated /t/ for /ð/.

It is interesting to note that Korean also lacks word-initial and final consonant clusters. Some learners will therefore tend to insert a short /u/ sound between consonants as a means to convert the English syllable structure into a Korean syllable structure (e.g., “snake” becomes /suneɪg/). In addition, Korean also has very few word-final consonants. For English words ending in an affricate, some learners will insert the sound /ɪ/ at the end of the word (e.g., “match” /mæʧ/ becomes /mæʧɪ/).

While there are significant differences at the segmental level, Korean and English also differ along suprasegmental dimensions. English is generally characterized as a stress-timed language while Korean is described as syllable-timed [9,10]. This difference makes it more difficult for Korean learners to achieve native-like English rhythm. Furthermore, native speakers of English can perceive learner speech as sounding boring or monotonous and difficult to comprehend since Korean does not utilize word or syllable stress features [7].

During the development of ReFLEX we found a significant body of literature describing the typical error patterns made by KLEs. Often, however, the relative frequency at which learners made such pronunciation errors was less well understood. To better understand error frequency, we collected and phonetically annotated a pilot corpus consisting of prompted English speech data from an assortment of different types of content. The corpus includes minimal pairs (e.g., right/light), stress minimal pairs (e.g., con<u>t</u>ent/con<u>t</u>ent), short paragraphs of text, sentence prompts, isolated loan words, and words with particularly difficult consonant clusters (e.g., refrigerator). In total, we collected 25,000 total speech samples from 111 learners who reside in Korea (55 beginner, 33 intermediate, and 23 advanced learners). Three human annotators conducted phone-level annotation. The corpus provides a direct comparison of realized phone sequences compared to expected canonical sequences from native speakers. To date, 15 speakers have been inter-annotated and another 15 speakers are intra-annotated to ensure reliability and accuracy of the annotations. Table 1 summarizes the most frequent segmental errors observed (error count divided by total errors).

TABLE I. Frequent KLE Segmental Error Patterns

Error Pattern Description	***Error %***
Unstressed to Stressed Vowel (e.g., /i/ → /ɪ/)	14%
/ɹ/ Issue (deletion, substitution, etc.)	11%
Dental to Alveolar (e.g., /θ/ → /t/)	9%
Consonant Deletion (word initial /j/ → / /)	8%
Vowel Insertion (e.g., large → largi)	7%
Diphthong to Monophthong (e.g., /eɪ/ → /ɛ/)	5%
Consonant Cluster Simplification	5%
Vowel to Central Vowel (e.g., /u/ → / i/)	5%
Vowel Deletion (e.g., /ə/ → / /)	4%
/æ/ to /ɛ/	4%

III. ROSETTA STONE REFLEX

A. System Overview

Rosetta Stone ReFLEX was designed for English learners with intermediate to advanced knowledge in syntax, grammar, and vocabulary but who lack the confidence and communicative skills to engage in natural conversation with native speakers. Specifically, the program aims to improve comprehensibility, automaticity, and confidence in conversational dialogue. To achieve this goal, the online software introduces the concept of an adaptive 30-minute daily regimen that focuses on three distinct areas: sound skills, rehearsed conversational dialogues, and live, one-on-one conversation practice with a tutor who is a native speaker of English. We refer to these areas as Skills, Rehearsal, and Studio, respectively, and will describe each component in the following sections.

B. Skills

As noted earlier, the ability for a learner to both perceive and produce challenging sound contrasts is important to achieving comprehensible speech in dialogue interaction. Several studies have shown that perception training is not only possible but also can lead to improvements in speech production [11]–[14]. During each daily program, learners are provided approximately five minutes of fun and engaging game-like activities that aim to improve perception and production of such contrasts. Figure 1 illustrates a word-sorting perception activity in which the learner must select minimal-pair words (shown right) and place them within corresponding contrast positions on the grid (shown left).

The core activities are designed using a grid-based framework in which a finite set of objects and their actions are carefully designed to ensure consistent UI interaction. Within the framework, circles are used to represent individual speech sounds, squares are used to represent whole words, and rectangles are used to represent sentences. Color is used as a final dimension to denote sound contrasts. Users can mouse-over on UI elements to play audio from each element.

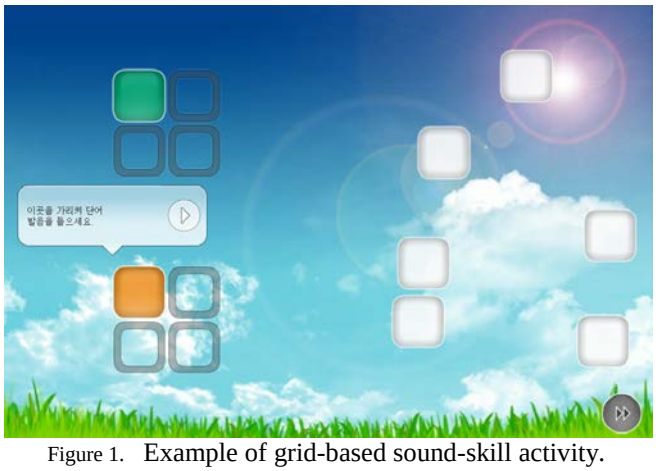


Figure 1. Example of grid-based sound-skill activity.

C. Rehearsal

The second key component of Rosetta Stone ReFLEX, known as Rehearsal, engages the learner in spoken dialogue conversation practice. Implemented within the context of a 3-D virtual world, learners practice conversational situations using speech recognition to receive immediate feedback. The main goal of Rehearsal is to strengthen automaticity—the ability to react naturally and quickly when confronted with common dialogue situations. The scenarios are based on learning core sentence and phrase templates that learners can use to generalize to many types of conversational situations. Example dialogue scenarios include situations such as ordering coffee at a café, asking for directions when lost, and asking for assistance on an airplane flight.

Figure 2. Rosetta Stone ReFLEX Rehearsal conversational dialogue practice.

For each scenario the learner is first provided a dialogue preview in which the conversation is played through using recordings of native speakers. Next, the learner plays the role of one of the characters by repeating the expected response for each dialogue act. Once the learner has practiced the conversation, the learner must perform the dialogue without the aid of prompts. An underlying adaptive engine keeps track of phrases that the learner has mastered and delivers appropriately challenging content. Depending on the dialogue's state, the learner may be provided alternative ways of responding within the situation. These communicative templates are at times reused and extended both within and across dialogue scenarios to reinforce the types of freedoms speakers have when engaging in a dialogue. Thus, learners may be experiencing an entirely new dialogue and quickly realize the templates they have previously learned are useful for their current situation.

The Rehearsal component is constantly adapting to the learner's performance. Each conversational scenario is implemented using a directed acyclic graph structure. Each miniconversation is represented as a subsequence of nodes within the entire graph structure. Based on user performance, the Rosetta Stone ReFLEX system can provide more or less variation of the content that the learner receives.

Rehearsal, like Skills, minimizes the usage of on-screen text as a means to enforce listening and speaking skills and to attempt to break the learner from using text as a crutch in learning English. During dialogue interaction in Rehearsal,

speaking is emphasized, but text is shown if the user makes multiple incorrect speaking attempts. Rehearsal also provides an in-line phrase practice mode that can be automatically invoked if the learner is struggling with speaking a phrase. This mode allows a long phrase to be broken down into smaller parts that can be practiced individually and then brought together to master the entire phrase.

Rehearsal makes exclusive use of speech recognition technology to make utterance verification decisions. The system must carefully balance the need to minimize false negatives (rejecting correctly spoken phrases) against the need to minimize false accepts (accepting incorrect responses). One key challenge here involves not only detecting out-of-grammar responses, but also providing appropriate feedback for poorly spoken responses.

D. Studio

Rehearsal sessions last approximately 15 minutes. During the remaining 5–8 minutes of the 30-minute daily session, the learner is connected via video for live one-on-one conversation practice with an American tutor (called a Studio Coach). The video stream of the coach is transmitted to the learner's computer, while the learner has the option of communicating via audio only or by video via webcam. Each Studio session reinforces content similar to what the learner has just practiced during Rehearsal. The Studio Coach has the flexibility to provide personalized pronunciation feedback as well as encouragement. For advanced learners, the Studio Coaches may alter the conversation script to help reinforce learners who try unanticipated phrases.

Figure 3. Example Studio interaction. Studio Coach is shown in the upper left of the learner's screen.

The Studio component is perhaps the most distinctive aspect of the Rosetta Stone ReFLEX system. It adds a human element to the service and provides learners with a strong sense of confidence that they can communicate with a native speaker. Aside from its value to the learner, Studio is possibly one of the most challenging components both from a technical and business perspective. Studio Coaches must be available 24–7 and especially at peak times when learners wish to use the online service. Learners must also be connected quickly and seamlessly to the Studio Coach when they complete their Rehearsal session. Technical issues such as low-delay audio and video as well as maintaining high audio quality are key challenges in providing such a service between KLEs and American Studio Coaches.

IV. UNDERLYING SPEECH TECHNOLOGIES

While Rosetta Stone ReFLEX shares some similarities with our TOTALe language-learning solution, it differs along several fundamental dimensions and contains several new and advanced speech technologies at its core. Rosetta Stone ReFLEX is designed to support advanced speech and user data logging as well as to deliver A/B contrast experiences for subsets of learners. This aspect allows the service to be reactive to learner performance, deliver personalized content to remedy issues, and allow for new activities and experiences to be trialed within user populations. Rosetta Stone ReFLEX incorporates Rosetta Stone's proprietary speech recognition technology. In Rosetta Stone ReFLEX, the speech recognition engine executes within Adobe Flash, providing a zero-install speech experience. Remote servers also process audio captured by the speech recognition engine to provide a detailed view of pronunciation error patterns for each learner. New pronunciation error detection systems can be deployed on the server side while new speech recognition engines and speech models are deployed as web content. Unlike cloud-based speech recognition services, running speech recognition within the web browser ensures real-time reaction and guaranteed availability of service. In the following sections, the core speech recognition technology is described, and our methodology for pronunciation error detection is summarized.

A. Core Speech Recognition Technology

Rosetta Stone ReFLEX is an online product delivered through a client computer's web browser using Flash technology. Historically, developers wishing to incorporate speech recognition via Flash needed to either connect to a client-installed native-speech engine (e.g., via TCP sockets) or stream audio to a remote computer (e.g., Flash supports the RTMP protocol using Flash Media Server). Recently, Adobe has released a tool known as *Alchemy* that makes it possible to compile C and C++ code to execute directly within an ActionScript Virtual Machine (AVM2) inside the Flash Platform [15]. Coupled with Flash's recent feature extension that allows access to sound samples captured from the client-side microphone, the two advances make it possible to run advanced speech recognition technologies seamlessly within the Flash environment. The advantages of using the Alchemy approach are severalfold. First, speech recognition essentially requires no installation on the client-side computer. Users simply need to allow Flash to access the computer's microphone. Second, knowledge sources such as acoustic models, pronunciation lexicons, or parameter settings essentially become web-based resources. Acoustic models, for example, are downloaded once to the client computer and stored within the browser's cache. Unlike cloud-based speech recognition technology, the client-side implementation through Flash ensures that speech recognition will always be available to the customer and will provide real-time feedback at all times. Finally, since updates to speech-related knowledge sources happen at the server side, the technology does not require users to explicitly install speech components

on their client computer. This technology has the added benefit that users always have the latest speech engine, speech models, and tuned application parameter settings.

It is important to point out that one current disadvantage to using Alchemy is process execution speed. On average, C/C++ code executed within Flash operates approximately 10–20 times slower than native client C/C++ code. Speech recognition systems need to be well optimized for efficiency to maintain real-time speed. This optimization can be quite tricky to achieve for pronunciation scoring systems given the breadth of low-end processors that are common today.

Despite the computational limitations of Alchemy, the proprietary speech technology deployed within Rosetta Stone ReFLEX is rather advanced. The speech engine incorporates front-end acoustic noise reduction by real-time tracking of background noise and performing suppression efficiently on Mel-scaled filter-bank energies. We found noise suppression to be incredibly important for such a real-world system, given the wide variety of microphones used in the wild. Our speech engine also incorporates nonnative HMM-based acoustic models for utterance verification as well as native acoustic models for performing pronunciation scoring. The application of both acoustic models happens in real time while the user speaks to provide a simultaneous decision regarding utterance acceptability and pronunciation quality. In addition, we perform incremental adaptation on the nonnative acoustic models to allow the speech engine to adjust itself to the accent level of the user. Thus, utterance verification essentially makes use of nonnative (L1-L2) knowledge as well as user-specific voice characteristics while pronunciation scoring is constrained to comparison with native-model acoustics.

In addition to performing client-side speech recognition (utterance verification and pronunciation scoring), the Rosetta Stone ReFLEX speech engine performs real-time audio compression and creates utterance-level data logs that are transmitted to the Rosetta Stone ReFLEX servers. This allows more advanced algorithms to be applied, for example, to compute detailed descriptions of the type and frequency of pronunciation errors made by the learner and to allow the system to provide appropriately personalized content.

B. Pronunciation Error Modeling

Our approach to pronunciation error detection and identification is based on modeling the phonological errors in L2 data using a machine translation (MT) framework [16]. Specifically, we treat the differences between native/canonical phone sequences and those produced by a nonnative learner as being describable as a translation process. For a given native phone sequence (source language), the best nonnative phone sequence (target language) that represents a good translation of the input can be described as

$$P(NN \mid N) = \arg\max_{NN} P(N \mid NN) \cdot P(NN) \qquad (1)$$

where N and NN are the corresponding native and nonnative phone sequences. P(N|NN) is considered the translation model. It characterizes the phonological transformations between the native and nonnative phone sequences. P(NN) is a language model for the nonnative phone sequences. It captures the likelihood of a certain nonnative phone sequence occurring in labeled L2 data.

A parallel phone corpus of canonical and annotated nonnative phone sequences is run through Giza++ [17] to obtain phone alignments. The phone alignments from Giza++ are loaded into the Moses decoder [18] to grow the one-to-one alignments into phone-chunk based alignments. This process is analogous to growing word alignments into phrasal alignments in traditional machine translation. The resulting phonological error model has phone-chunk pairs with differing phone lengths and a translation probability associated with each one of them. A nonnative 3-gram phone language model is trained using IRSTLM toolkit [19] by feeding in annotated phone sequences from the L2 data. Given the phonological error model and a nonnative phone language model, the Moses decoder can generate the N-best nonnative phone sequences for any given canonical native phone sequence.

The MT approach to phone-error modeling offers several distinct advantages over rule-based approaches. First, the MT approach automatically learns all phenomena that consistently occur in the annotated data. This includes language-transfer effects and other phenomena like mispronunciations caused by interference due to word orthography. For example, nonnative speakers of English often pronounce the “b” in the word “debt”. The MT approach also allows for one-to-many, many-to-one, and many-to-many alignments that allow for better modeling of phonological error phenomena that span across phone chunks. This can be especially effective in modeling severe insertion or deletion problems spanning across phoneme subsequences. For example, KLEs often mispronounce the word *refrigerator* (pronounced /ɹəfɹɪʤ əɹeɪɾɚ/) as /lipɹɪʤ iɾeɪɾɚ/. The MT system has the ability to learn the correspondence between the native phone chunk “ɹəf” and the nonnative chunk “lip”. Current approaches to phonological error modeling lack a strong mathematical framework that would facilitate auto-generation of nonnative phone sequences. Most of these systems employ heuristic-based rule selection and application techniques (as rule interdependencies are not explicitly modeled). The decoding paradigm that the MT approach offers is a much more principled way of combining error-rule probabilities and interdependencies between error rules to generate the most probable nonnative phone sequences for error detection.

C. Pronunciation Error Detection

For server-side pronunciation error detection, we utilize the phonological error model and nonnative phone language model to automatically generate nonnative alternatives for every native pronunciation. We use a 4-best list to strike a good balance between under-generation and over-generation of pronunciation alternatives. The generated nonnative lexicon (which includes canonical pronunciations) along with an American English acoustic model is used to recognize the spoken utterance. For utterances that pass verification, a Viterbi alignment of the audio and the expected text is performed. The search space is constrained to the native and nonnative variants of the expected utterance. The phone sequence that maximizes the Viterbi path probability is then

aligned against the native/canonical phone sequence to extract the phonological errors produced by the learner. On the KLE speech corpus described in Section II, we found the MT approach achieves a 32.9% relative improvement (F-1 score) in phone error detection and a 49% relative improvement in phone-error identification compared to a traditional edit-distance based approach to modeling phone errors. Details can be found in [16].

IV. Practical Challenges and Research Directions

Since launching Rosetta Stone ReFLEX in Korea in July of 2011, we have captured nearly 7 million spoken utterances from several thousand learners. One practical research challenge relates to simply analyzing and aggregating meaningful learner metrics over such significantly large data. Striking a balance of what to relate back to the learner without overwhelming them as well as factoring out influences such as background noise are two example challenges. Practically speaking, acoustic scores can be influenced simply by microphone sound quality. Tracking and monitoring microphone sound quality (e.g., using measures such as signal-to-noise ratio, peak-clipping detection, etc.) and understanding its impact on learner metrics is of practical importance. In terms of research directions, we see great potential for incorporation of stress, rhythm, and intonation activities into Rosetta Stone ReFLEX and have made initial progress toward developing and defining corresponding metrics [20,21]. We have also recently extended the MT framework for pronunciation error detection to the problem of predicting word-level errors made by learners [22] and have extended our pronunciation error detection framework using discriminatively trained nonnative acoustic models [23]. For academic researchers, defining improved metrics that can holistically describe the degree of nativeness in terms of segmental and suprasegmental qualities seems to be an area that could contribute to future success in language-learning systems such as Rosetta Stone ReFLEX.

References

[1] J. H.-Chan, "The Economics of English," Samsung Economic Research Institute, Seoul, Nov. 2006.

[2] B. Lee, Korea's endless grapple with English, *Korea Joongang Daily*, Feb, 2008.

[3] T. O'Donnell, "Learning English as a foreign language in Korea: Does CALL have a place?" *Asian EFL Journal*, vol. 10, April 2006.

[4] S. Choi, "Teaching English as a Foreign Language in Korean Middle Schools: Exploration of Communicative Language Teaching through Teacher's Beliefs and Self-Reported Classroom Teaching Practices," Ph.D. dissertation, Ohio State University, Columbus, Ohio, 1999.

[5] T. Adams, *Linguavore* [blog], "Introducing Rosetta Stone Version 4 TOTALe: Read, Write, Speak, and Understand," Sep. 15, 2010. Available: http://blog.rosettastone.com/introducing-version-4-totale/

[6] P. Avery, S. Ehrlich, *Teaching American English Pronunciation.* New York: Oxford University Press, 1992.

[7] J. Lee, Korean speakers. In Smith, B. and Swan, M., Eds., *Learner English: A Teacher's Guide to Interference and other Problems*, 2nd ed. (Cambridge Handbooks for Language Teachers). New York: Cambridge University Press, 2001, pp. 296–309.

[8] J. Cho, H-K. Park, "A comparative analysis of Korean-English phonological structures and processes for pronunciation pedagogy in interpretation training." *Translators' Journal*, vol. 51, no. 2, 2006, pp. 229–246.

[9] R. Dauer, "Stress-timing and syllable-timing reanalyzed." *Journal of Phonetics* 11, 1983, pp. 51–62.

[10] T.-Y. Jang. "Rhythm metrics of spoken Korean." *Language and Linguistics* 46, 2009, pp. 169–186.

[11] J. Logan, S. Lively, and D. Pisoni, "Training Japanese listeners to identify English /r/ and /l/: A first report," *J. Acoust. Soc. Am.* 89: 874–886, 1991.

[12] R. Akahane-Yamada, Y. Tohkura, A. R. Bradlow, D. B. Pisoni, "Does training in speech perception modify speech production?" in *Proc. of ICSLP*, 1996.

[13] A. R. Bradlow, D. B. Pisoni, R. Akahane-Yamada, and Y. Tohkura, "Training Japanese listeners to identify English /r/ and /l/: IV. Some effects of perceptual learning on speech production," J. Acoust. Soc. Am., 101(4): 2299–2310, 1997.

[14] D. Nobre-Olveira, "Effects of perceptual training on the learning of English vowels in non-native settings" in *New Sounds 2007*: *Proceedings of the Fifth International Symposium on the Acquisition of Second Language Speech*, Florianópolis, Brazil, 2007.

[15] Adobe Labs. (2012, Mar. 23). *Alchemy*. Available: http://labs.adobe.com/technologies/alchemy/

[16] T. Stanley, K. Hacioglu, B. Pellom. "Statistical machine translation framework for modeling phonological errors in computer assisted pronunciation training system," *Proceedings of Interspeech*, Florence Italy, 2011.

[17] F. J. Och, H. Ney, "A systematic comparison of various statistical alignment models," *Computational Linguistics*, Vol. 29, No. 1, March 2003, pp. 19–51.

[18] P. Koehn et al., "Moses: Open source toolkit for statistical machine translation," Annual Meeting of the Association for Computational Linguistics (ACL), Prague, Czech Republic, June 2007.

[19] M. Federico, N. Bertoldi, M. Cettolo, "IRSTLM: an open source toolkit for handling large scale language models," *Proceedings of Interspeech*, Brisbane, Australia, 2008.

[20] M. Mehrabani, J. Tepperman, E. Nava, "Nativeness Classification with Suprasegmental Features on the Accent Group Level," Submitted to Interspeech 2012, Portland Oregon, unpublished.

[21] J. Tepperman, T. Stanley, K. Hacioglu, B. Pellom, "Testing suprasegmental English through parroting," in *Proceedings of Speech Prosody*. Chicago, USA, May, 2010.

[22] V. Siivola, K. Hacioglu, "Modeling typical word level errors of learners of a new language," Submitted to Interspeech 2012, Portland, Oregon, unpublished.

[23] T. Stanley, K. Hacioglu, "Improving L1-specific Phonological Error Diagnosis in Computer Assisted Pronunciation Training," Submitted to Interspeech 2012, Portland, Oregon, unpublished.

Automatic Assessment of Non-Native Prosody – Annotation, Modelling and Evaluation

Florian Hönig, Anton Batliner, and Elmar Nöth
Pattern Recognition Lab
Universität Erlangen-Nürnberg
Martensstr. 3, 91058 Erlangen, Germany
Email: {hoenig,batliner}@informatik.uni-erlangen.de

***Abstract*—The first part deals with general considerations on the evaluation of both human raters and automatic systems, employed for pronunciation assessment: How can we come closer to an unbiased, realistic estimate of their reliability, given the fallibility of human annotators and the nature of machine-learning algorithms (and researchers) that adapt, and inevitably overfit to a given training set. In the second part, we will present concrete models for the assessment of the overall rhythmic quality of the learner's speech. The methods are evaluated in detail on read and semi-spontaneous English data from the German research projects C-AuDiT and AUWL.**

I. Introduction

A. Motivation

Non-native prosodic traits limit proficiency in a second language (L2) and by that, mutual understanding. Prosodic phenomena, located on word level and above, encompass word accent position, syntactic-prosodic boundaries, and rhythm, and help listeners to structure the speech signal and to process segmental, syntactic, and semantic content successfully. Non-native prosodic traits are therefore not mere idiosyncrasies, but often seriously hamper mutual understanding. Thus, they have to be modelled in computer-assisted pronunciation training (CAPT).

A few studies deal with non-native accent identification using prosodic parameters [1]–[3]. In [4], the automatic detection of erroneous word accent positions in English as L2 is addressed. Suprasegmental native traits have been, e. g. investigated recently in basic when trying to model language-specific rhythm [5], [6]. Maybe the most important general factor to be modelled in CAPT is non-native rhythm: the English prosody of, e. g. French, Spanish, or Hindi native speakers can sound 'strange'. The reason is a difference in rhythm that has been noted amongst others by [7], p. 97, who speaks about syllable timed languages such as French ("the syllables [...] recur at equal intervals of time – they are *isochronous*"), and stress-timed languages such as English ("the stressed syllables [...] are isochronous"). [5] and [6] challenge this traditional terminology because in empirical studies, such an isochrony could not be observed; they claim that it is rather a more complicated constellation where especially syllables not carrying the word accent, that are weak (schwa) in 'stress-timed' languages, are produced stronger in 'syllable-timed' languages. Thus we might expect such differences to show up in L2 learners whose native language L1 does not display the native structure of L2.

To cope with these traits, L2 teachers can use explicit feedback, i. e. denote the very pronunciation error, or implicit feedback, i. e. repeat (parts of) lessons which proved to be difficult for the learner. The same strategies are available for Computer-Assisted-Pronunciation-Training (CAPT) programs. Explicit feedback should be used – but only if there is a high recall and a low false alarm rate. However, we are still far from any 'perfect' *localization* of pronunciation errors; other things being equal, a *global* assessment (of sentences, paragraphs, or whole sessions) has higher chances to correctly indicate (types of) coarse errors the learner tends to make. If any localised assessment is available, we can use this information for giving both explicit and implicit feedback, whereas a global assessment implies the sole use of implicit feedback. Rhythm is insofar special as it is often difficult to pinpoint exactly what's wrong with a given utterance that sounds unnatural, and even more difficult to convey to the learner what exactly to improve. Thus, implicit feedback to the learner's rhythm in CAPT programs, e.g. in a say-after-me or speak-with-me (shadowing) manner may be a good option for pedagogical reasons, too.

B. Machine Learning: Evaluation

The pattern recognition approach – i. e. collect annotated data, extract suitable features, and train a supervised classification or regression module using machine learning – is a universal and powerful tool in CAPT. However, great care has to be taken when estimating the accuracy of such an approach: If the collected data are not *representative* of the intended application, a strict division into *training* and *test set* has to be used during evaluation. Otherwise, the used algorithms and choices taken by the researcher may *overfit* to the data and yield optimistic estimates of the accuracy [8, p. 19]. In CAPT research (and many other fields), the data are usually *far* from being representative, as there is no running application (yet) to draw data from, and eliciting, collecting and especially annotating is expensive. Moreover, linguistic data are *per se* an open set.

Even if the manifold pitfalls of overfitting to the data[1]

[1] feature selection, accidentally tuned to the whole dataset, may stand as a popular example

are avoided, and the evaluation is technically sound in so far as all items (*instances*) used for testing have never been used for training/tuning, the estimated accuracy may still be meaningless. The reason is that just keeping training and test set disjunct w. r. t. the *instances*, and mixing everything else[2], is not enough. In fact, each partition into training and test has to be designed in such a way that the test reflects the conditions in the eventual application. For example, in the speech recognition community it is widely recognized that train and test have to be disjunct w. r. t. speakers in order to arrive at realistic estimates of the accuracy.

Depending on the application, there may be other conditions that should be different, too. When there is more than one condition that needs to be different in training and test, the evaluation gets wasteful: either time-consuming nested cross-validation schemes have to be used or only a fraction of the hard-won data can be utilized. This becomes a problem when organizing competitions such as the INTERSPEECH 2009 Emotion Challenge [9] where cross-validation is not a practicable option. A way to avoid this problem from the start would be to collect data in (at least four) partitions that are designed in such a way that they are mutually independent w. r. t. all conditions.

For CAPT, one usually wants to employ a module that works well not only for unseen speakers, but also for unseen material [10]. We will therefore include in our evaluation not only speaker-independent settings, but also a setup where train and test is disjunct w. r. t. both speakers and sentences – this will have dramatic consequences on the resulting accuracy.

C. Generative vs. Discriminative Approaches

For evaluating the accuracy of a CAPT method, we will always need a body of data from non-native speakers that includes annotated examples of both good and bad speaking performance. For modelling, however, two basic approaches can be identified:

Generative or indirect: The model only describes what is *acceptable*, and a distance measure is used to derive a score or to decide for 'correct' or 'error'. The advantage is that data collection is far easier: we can use native speech, and more importantly, when neglecting the few errors that native speakers make as well, we do not need error annotations. For example, when applying the Goodness of Pronunciation (GOP) measure [11] to identify mispronounced phonemes, we can use just transcribed native speech to build models for correctly produced phonemes, and use (an approximation of) the a posteriori probability of the target phonemes as a similarity measure.

Discriminative or direct: The model describes *both* acceptable and unacceptable pronunciations, and the pronunciation score or the decision 'correct' or 'wrong' is a direct output of the classification or regression module. This approach has the potential for optimal accuracy but data collection is much more expensive, as enough annotated non-native speech comprising both good and bad pronunciations is needed, i. e. much much more than for the evaluation of generative approaches. For the example of detecting mispronounced phonemes, this is practically infeasible in the general case due to data sparsity resulting from coarticulation effects and the different L1s of the targeted learners. For modelling frequent errors of certain target speaker groups however, it may be the method of choice, e. g. /θ/ → /s/ for German learners of English as L2.

In practice, both approaches are often mixed to reach satisfying accuracy with feasible effort, e.g. a generative model for correct phonemes is used but a priori probabilities for mispronunciations of the target group of speakers are included.

Assuming that modelling pronunciation quality w. r. t. rhythm is less complex than modelling segmental pronunciation, we followed the discriminative approach in the present work.

D. Annotation: General Considerations

As discussed above, we need to establish reference scores for evaluating, and possibly also for training our pronunciation scoring method. Apart from the speech data that should be annotated – type, size, (balanced, stratified) sub-samples such as male/female, degree of proficiency, etc. – the main alternatives to be chosen from is a choice between experts and 'naïve' subjects for annotation and/or perceptive evaluation, and the decision on how many people to employ for the annotation task.

1) Labeller Agreement and Multiple Labellers: The variability between labellers can be traced back to at least two main factors: first, labeller-specific *traits* such as gender, dialect, sociolect, talent for assessing speech, etc., and second, speaker-specific *states* such as boredom, interest, tiredness, illness, etc. Together, all these factors can be modelled as error whose variability is higher if less subjects are employed. Following this logic, we can define the *ground truth* as the average over infinitely many labellers (for a certain group of labellers).

How many labellers to actually employ is foremost a matter of time and money – as long as some rules of thumb are followed: if there are three or more labellers, we can use majority decisions. If there are five or more labellers, we are more safe when establishing quasi-continuous judgements from ordinal ones, based on the average score of all annotators. Intuitively, around 10 is a good figure; more than 20 are employed rather rarely. In our own experience, we found that for the task of rating prosody of L2 English speech on a continuous scale, 10 labellers already yield a very good reference: A reference A^N created by averaging over the (normalised) annotations of N labellers with an average pairwise Pearson correlation of c can be expected to be correlated to the ground truth as follows [12]:

$$\mathrm{Corr}(A^N, A^\infty) = \sqrt{c/(\frac{1}{N} + \frac{N-1}{N}c)}. \quad (1)$$

Thus, despite of a low pairwise correlation of 0.3, averaging over 10 labellers already yielded a reference with a correlation

[2]as is commonly done when doing cross-validation with machine learning packages

of 0.90 to the ground truth.

2) Expert vs. Naïve Labellers: Experts being able to do a detailed annotation are rare and more expensive than naïve raters; moreover, they may be biased in some way towards their own theoretical preferences. Naïve subjects are less expensive, thus more of them can be employed, and they are less biased, but care has to be taken that the task is well-defined; moreover, we cannot expect them to be as consistent and competent as the experts. Normally, less experts are employed than naïve subjects. So far, however, there are no strict guidelines for that; recently, there seems to be a trend towards low-cost (non-expert) crowdsourcing using, for example, Amazon Mechanical Turk [13]: Snow et al. conclude that for the task of affect recognition in speech, using non-expert labels for training machine learning algorithms can be as effective as using gold standard annotations from experts. Also in [14], it was shown that a large number of annotators ('Vox Populi') creates reliable annotations.

In our experiments on rating L2 German speech with respect to prosody, we compared three groups of native labellers with different expertise: naïves, phoneticians, and phoneticians with extensive labelling experience with the actual database ('real' experts) [15]. As expected, the consistency rose with the level of expertise: Regardless of whether we aim at the ground truth of experts, phoneticians or naïves, when only one labeller is employed, an expert is always the best choice and a naïve labeller the worst choice. However, when employing more labellers, good correlations to any of the three 'ground truths' can be achieved by all labeller groups. If c is d are the average pairwise correlation within two groups $\mathbb{A}$ and $\mathbb{B}$, respectively, and e the average pairwise correlation between a pair of labellers from the two groups, the correlation of N averaged labellers A^N from $\mathbb{A}$ with the ground truth B^∞ of $\mathbb{B}$ can be expected to be

$$\mathrm{Corr}(A^N, B^\infty) = \frac{e}{\sqrt{\frac{1}{N} + \frac{N-1}{N} c} \cdot \sqrt{d}}. \qquad (2)$$

Thus, we observed in our German data when employing at least five labellers of any of the three groups, very good references result that can be expected to be correlated to the ground truth of any of the three groups with at least 0.94.

3) Different L1 Dialect Backgrounds: Another aspect that may influence the perception and thus the rating of non-native speech is the background of the labellers with respect to their L1 variety. For our English data, we compared native speakers of American, British and Scottish English [16]. We observed slight differences in the perception of non-native accent, but practically no difference in scores for intelligibility or prosody. Thus we can speculate that irrespective of their own dialect or regional accent, annotators have internalised a common standard of their own L1.

4) Correlated Scores: When collecting labels for different scales such as intelligibility and prosody, one will usually find that the ratings are correlated among themselves to some extent. In this context, it is desirable to abstract from individual labeller variability. We can do this by estimating the correlation between the ground truths of the scales: if c and d are the average pairwise correlations within the labels for each of the two scales, and e the average correlation between one labeller's first scale with another labeller's second scale, we can use Equation 2 with $N \to \infty$, i. e. $e/\sqrt{cd}$.

5) Weighting Labellers: Even within a homogeneous group of labellers there will be individual differences regarding talent, diligence, time spent on task etc. which will have an impact on the quality of the annotations. Therefore an obvious possibility to improve the quality of the reference is to assign weights when averaging multiple annotations. A basic and robust approach is choosing the correlation to the other labellers as a weight [17]. In [18], a maximum likelihood estimator is derived that estimates weights and the combined reference in an iterative manner. In our own experiments, we use a similar but simplified, and more stable approach: using initially uniform weights, we estimate the mean square error of the labeller w. r. t. the ground truth, and set the weight indirectly proportional. On our German and English prosody scores, however, we did not see a big improvement by using weights, with neither of the three mentioned methods. For example on our new semi-spontaneous dialogue data (see below), we could improve the expected correlation to the ground truth just a little bit from 0.85 to 0.86. A reason for this may be that we hired our labellers in the traditional way with personal contacts, and paid them well, so we did not encounter problems that are reported for using e. g. Amazon Mechanical Turk where one has to check for 'spammers' who try to get the money without doing any real annotation.

6) Paying Labellers: When planning the annotation of a database, it is very convenient to pay piece-work (i. e. per annotated time, and not per annotation time). Often the money available for annotation is fixed, and then one can calculate what portion of the data one can afford to have annotated with how many labellers. To be ethically acceptable, the payment should not be too low; still, the quality of the annotation may suffer because some labellers may try to finish the job as quickly as possible. Other labellers take more time; paying them the same is unfair twice, as the slower labellers tend to deliver higher quality. Another possibility would be to pay per quality, where a quality measure could be derived in the same way as the weights when creating a combined reference by weighted averaging. In the annotation during the AUWL project, we observed a correlation of 0.87 between weights (assigned for intelligibility, non-native accent and prosody) and the time spent by the labellers. Although we cannot call this significant (only five labellers for this task), this is a strong trend. Nevertheless, it is still questionable to use that as a basis for payment: if the quality measure is normalized with respect to all labellers, labellers will effectively compete against each other, and if the quality measure is absolute, payment will be less when the task is difficult. Summing up, the best way still seems to pay an hourly wage, if organisational constraints allow for that.

In the annotation for the AUWL project, we could only pay piecework. However, after completion, we realized that one

labeller – the best – took

almost twice as much time as the rest, so we decided to pay an extra compensation.

7) Comparing Humans and Machines: For judging the estimated accuracy of an automatic system, it is instructive to compare with human performance. In order to do that in a fair manner, one should be aware that the correlation of the system with the reference is at best an optimistic estimate of accuracy. In the end, we want to know how well a system predicts a certain (abstract) *score*, not the collected imperfect *reference*, so we can for example not claim that any system performs better than the combined labellers. In fact, the final accuracy of a system should be estimated as the correlation between system output Y and ground truth A^∞:

$$\mathrm{Corr}(Y, A^\infty) \quad = \quad \mathrm{Corr}(Y, A^N) \cdot \mathrm{Corr}(A^N, A^\infty). \quad (3)$$

Consider for example a hypothetical system that is trained with the average of five labellers with a pairwise correlation of 0.5. Using Equation 1, this yields a reference with an expected correlation to the ground truth of ≈ 0.91. At the first glance, an automatic system that correlates with the reference with 0.6 seems better than the average human. However, as argued above, the correlation of the system with the ground truth can only be expected to be $0.6 \cdot 0.91 \approx 0.55$. On the other hand, a single labeller can be expected to correlate with the ground truth with $\sqrt{0.5} \approx 0.71$. Thus, we should be careful not to underestimate human performance or overestimate the performance of our systems.

II. Databases

For the present work, we use two databases: Read English material from our German research project C-AuDiT and the EU project ISLE [19], and new semi-spontaneous English data from research project AUWL, collected with the help of our dialogue training tool *Dialogue of the Day* (dod).

A. C-AuDiT

Read material is, of course, less naturalistic than spontaneous one; however, it has two advantages: First, it is easier to process, and second, it allows incorporation into existing automatic training software which still builds upon written and read data. Thus, it is a relevant object of study, also from the point of view of an commercial applicant of CAPT.

1) Material and Speakers: We recorded 58 English L2 speakers: 26 German, 10 French, 10 Spanish, 10 Italian and two Hindi speakers, and additionally 11 native American English (AE) 'reference' speakers. They had to read aloud 329 utterances shown on the screen display of an automated recording software, and were allowed to repeat their production in case of false starts etc. The data to be recorded consisted of two short stories (broken down into sentences to be displayed on the screen), sentences containing, amongst other, different types of phenomena such as intonation or position of phrase accent (*This is a house.* vs. *Is this really a house?*), or tongue-twisters, and words/phrases such as *Arabic/Arabia/The Arab World/In Saudi-Arabia, ...*; pairs such as *'subject* vs. *sub'ject* had to be repeated after the prerecorded production of a tutor. Where applicable, an expert annotated a likely distribution of primary and secondary phrase accents and B2/B3 boundaries [20] of a prototypical, articulate realisation.

When designing our recordings, we took 30 sentences from the ISLE database [19], which contains non-native English from 26 German and 26 Italian speakers. From this intersection, we defined the subset of the following five sentences that were judged as 'prosodically most error-prone for L2 speakers of English' by three experienced labellers [4]:

> *We're planning to travel to Egypt for a week or so.*
> *Can I have soup, then lamb with boiled potatoes, green beans and a glass of red wine?*
> *They will have to transport the components overland.*
> *The referee needed a police escort after the match.*
> *The company expects to increase its workforce next year.*

2) Annotation: Taking all speakers from C-AuDiT and ISLE that spoke all five sentences, we arrived at approx. one hour of speech from 94 speakers. Using the tool PEAKS [21], the annotation was conducted as a web-based perception experiment. Twenty-two native AE, 19 native British English (BE), and 21 native Scottish English (SE) speakers with normal hearing abilities judged each sentence in random order regarding different criteria, answering questions on intelligibility, non-native accent and the following two questions on prosody on a five-point Likert-scale:

- THIS SENTENCE'S MELODY SOUNDS...
  (1) *normal* (2) *acceptable, but not perfectly normal* (3) *slightly unusual* (4) *unusual* (5) *very unusual*
- THE ENGLISH LANGUAGE HAS A CHARACTERISTIC RHYTHM (TIMING OF THE SYLLABLES). HOW DO YOU ASSESS THE RHYTHM OF THIS SENTENCE?
  (1) *normal* (2) *acceptable, but not perfectly normal* (3) *slightly unusual* (4) *unusual* (5) *very unusual*

As already mentioned above, we found no real difference between the ratings from the AE, BE, or SE labeller, so we lumped them all together to get a single combined score for each utterance. It turned out that these combined ratings for *mel* and *rhy* are highly correlated among themselves with 0.95. So the question was whether the labeller were at all able to distinguish between the two. To answer this, we estimated the correlation between the ground truth of the scores as described in Section I-D4. Thus, we can expect the ground truths of *mel* and *rhy* to correlate with 0.97. Interestingly, we found the British labellers to behave a bit different (0.95; AE: 0.98, SE: 0.99). Our conclusion is that there may be a small difference, but too small to be considered for automatic assessment at the current state of the art. Thus, we decided for the present study to form a combined rating *pros* by averaging the 124 (normalized) annotations of both the *mel* and *rhy* scores. The expected correlation of this combined score with its ground truth is 0.99.

B. Dialogue of the Day (dod)

Reading leads to a special speaking style and can have a disruptive effect on speech, especially for learners with low L2 competence. Therefore, we took a different approach to data collection in our research project AUWL and designed a client-server tool for practising pre-scripted dialogues.

1) Training Tool: Before embarking on the dialogue training, the learner can first listen to the whole dialogue spoken by reference speakers. Then the learner enacts the dialogue with a reference speaker as a dialogue partner. In doing so, he can either have his lines prompted by a reference speaker and repeat afterwards, or directly read them off the screen (karaoke), or speak simultaneously with a reference speaker (shadowing). For facilitating shadowing, it can optionally be combined with prompting. Taking into account less proficient learners, one can choose between reference recordings spoken in a normal or in a slow tempo, and longer dialog steps can be subdivided. Options for choosing from different reference speakers, swapping roles, (re-)starting from an arbitrary position, replaying the latest own version of a dialog step, replaying the whole enacted dialog, or using own recordings for the dialog partner, complete the versatile training tool which is admittedly too complicated for end customers. Using this tool, we were able to elicit application-relevant speech which is considerably more spontaneous and less reading-style.

2) Material: We created 18 dialogues on topics such as business negotiations, shopping or holidays, with six for each of the CEF [22] levels A2 (elementary), B1 (pre-intermediate), und B2 (intermediate). Three female and three male professional native speakers spoke the material in both normal and slow tempo, resulting in 1908 recorded reference utterances or 2.2 hours of speech. As for the C-AuDiT material, we annotated a likely distribution of primary and secondary phrase accents and B2/B3 boundaries of a prototypical, articulate realisation for each dialogue. Possible points for subdivisions were annotated independently, because the presumptive B3 boundaries annotated were not suitable in some cases. The audio tracks to be replayed for the subdivided mode were created by automatically cutting the whole recordings, using a speech recognizer for segmentation.

3) Speakers: We started with 85 volunteering learners, who got a login for a web-based system where they could use the training tool in a self self-dependent manner. The learners were free in the choice of the dialogs and training modes such as shadowing. All recordings and dialogue timing information were stored at the server. Although we asked the learners to use a headset, the resulting audio quality was quite heterogeneous. At the end of the data collection, we got usable speech material from 31 speakers. According to a self-assessment, CEF levels are distributed as follows: 5×A2, 5×B1, 10×B2, and 11×C1. In total, the material amounts to 5145 utterances in 1019 dialog runs or 7.8 hours of speech. Each utterance was classified by a single annotator into 'clean' (5.5h), 'usable' (mainly usable content, but word editing and louder noise such as coughing; 1.7h), or 'unusable' (unusable content or dominant noise due to wrong audio settings etc.; 0.6h).

4) Annotation: The clean and usable material was annotated by five native post-graduate phoneticians. As with the C-AuDiT material, we asked questions on intelligibility and non-native accent on a five-point Likert scale, but according to our experience with the *mel* and *rhy* scores, we just asked one merged question regarding prosody (*pros*):

> THE ENGLISH LANGUAGE HAS A CHARACTERISTIC PROSODY (SENTENCE MELODY, AND RHYTHM, I.E. TIMING OF THE SYLLABLES). THIS SENTENCE'S PROSODY SOUNDS…
> (1) *normal* (2) *acceptable, but not perfectly normal*
> (3) *slightly unusual* (4) *unusual* (5) *very unusual*

We normalized and averaged the five annotations to get a single score for each utterance; for *pros* its expected correlation to the ground truth is 0.85. Additionally, the labellers had to mark words or parts of a sentence with a particularly unusual/non-native prosody. We measured the time the labellers took for annotation: we observed real-time factors between 2.2 and 7.5 (average: 3.9). For the present evaluations we only use the utterances classified as clean.

III. Prosodic Features

In order to obtain suitable input parameters for an automatic prosody assessment system, we compute a prosodic 'fingerprint' of each utterance. All processing is done fully automatic; however, we assume that the spoken word sequence is identical with the utterance the speaker had to read. First, the recordings are segmented by forced alignment of the target utterance using a cross-word triphone HMM speech recognition system. Then, various features measuring different prosodic traits are calculated. They are an extension to those described in [16] and adapted to utterance level instead of speaker level.

A. Specialized Rhythm Features

There is a body of research on modelling language-specific (native) rhythm. These hand-crafted, specialized parameters are promising candidates for our task.

1) Duration Features (Dur): A basic but fundamental property of speech is how fast something is said. We compute the average duration of all syllables of the utterance, and the average duration of vocalic and consonantal intervals (two features).

2) Isochrony Features (Iso): In order to capture possible isochrony properties [7], we calculate distances between centres of consecutive stressed or consecutive unstressed syllables. The centres are identified as the frames with maximal short-time energy within a nucleus. We compute six features: mean distances between stressed, and between unstressed syllables, standard deviations of those distances, and the ratios of those means and standard deviations.

3) Variability Indices (PVI): Following [5], we identify vocalic and consonantal intervals and calculate the raw Pairwise Variability Index (rPVI) which is defined as the absolute difference in duration of consecutive segments and its normalised

version nPVI (rPVI divided by the mean duration of the segments) for vocalic and consonantal segments. Additionally, following [23], we compute the control/compensation index (CCI) for vocalic and consonantal segments. This variant of rPVI takes into account the number of segments[3] composing the intervals. In total, six PVI features are computed.

4) Global Interval Proportions (GPI): Following [6], we compute the percentage of vocalic intervals (of the total duration of vocalic and consonantal intervals), and the 'Deltas': standard deviation of the duration of vocalic and consonantal intervals. Additionally, we include variation coefficients ('Varco') [24] for vocalic and consonantal intervals, i. e. normalized versions of the deltas. Together, we compute five Global Proportions of Intervals.

5) Combination of All Rhythm Features: Later in the experimental evaluation, these feature groups will either be analysed individually, or pooled (*Rhy-All*, 19 features).

B. General-Purpose Prosodic Features (Pros)

The expert-driven, specialized features described above are all based on duration, so they might miss other relevant information present in the speech data, such as pitch or loudness.

Therefore, we tried to capture as much potentially relevant prosodic information of an utterance as possible in an approach somewhere between knowledge-based and brute-force.

1) Local Features: We first apply our comprehensive general-purpose prosody module [25] which has proven suitable for various tasks such as phrase accent and phrase boundary recognition [25] or emotion recognition [26]. The features are based on duration, energy, pitch, and pauses, and can be applied to locally describe arbitrary units of speech such as words or syllables. Short-time energy and fundamental frequency (F0) are computed on a frame-by-frame basis, suitably interpolated, normalized per utterance, and perceptually transformed. Their contour over the unit of analysis is represented by a handful of functionals such as maximum or slope. To account for intrinsic variation, we include normalized versions of some of the features based on energy and duration, e. g. the normalized duration of a syllable based on the average duration of the comprising phonemes and a local estimate of the speech rate. The statistics necessary for these normalization measures are estimated on the native data of each corpus (11 native speakers amounting to five hours for *C-AuDiT*; 6 native speakers in two different tempi amounting to 2.2 hours for dod).

2) Global Features: We now apply our module to different units and construct global (utterance-level) features from that. Trying to be as exhaustive as possible, we use a highly redundant feature set (742 features) leaving it to data-driven methods to find out the relevant features and the optimal weighting of them. We compute:

- Average and standard deviation of the prosodic features derived from all *stressed syllables* (context '$0, 0$'), from all segments comprising stressed syllables and their direct successor (context '$0, +1$'), from all syllables succeeding stressed syllables (context '$+1, +1$'), and so on up to contexts '$-2, -2$' and '$+2, +2$'. The same is done for just the nuclei of stressed syllables. These features can be interpreted to generically capture isochrony properties inspired by [7].
- Average and standard deviation of the prosodic features derived from all words (context '$0, 0$'), and from all segments comprising two words (context '$0, 1$'). The same is done for syllables and nuclei. These features can be interpreted as generalizations of the deltas and proportions proposed by [6], [24].
- Average of the absolute differences between the prosodic features derived from consecutive units. This is done for contexts '$0, 0$' and '$0, 1$' of all words, syllables and nuclei. These features can be interpreted to generalize the pairwise variability indices proposed by [5], [23].

C. Combination of all Global Features

The combination of *Rhy-All* and *Pros* will be referred to as *All* in the evaluation.

IV. Modelling

The collected Likert scores for prosody are discrete random variables with five possible values. One option would therefore be to formulate the automatic assessment task as a five-class classification problem. However, we chose to automatically assess the pronunciation on a continuous scale, i. e. regression for the two reasons:

- When merging multiple labellers to get a reliable reference, information is lost, the more so as the ratings 'unusual'/'very unusual' are chosen rarely. Averaging the scores to get a quasi-continuous reference solves this problem.
- Classification does not reflect the ordinal nature of the labels.

A. Global Model

Our first approach was to take a number of utterance-level features as described in Section III and feed them, together with the reference values, to a suitable machine learning algorithm for regression. We chose Support Vector Regression (SVR), using WEKA [27]. With a suitably chosen complexity parameter C and kernel function, one can achieve both linear and non-linear models with good generalization ability even in the presence of many features.

B. Local Model

Our alternative approach uses a divide-and-conquer strategy: First, all syllables of an utterance are scored individually; the resulting scores are then combined by averaging[4]. For predicting a syllables score, we again apply SVR. Because we do not have a syllable level annotation, we use the score for

[3]Usually a phoneme is one segment; exceptions are e. g. long vowels which count as two segments.

[4]Our efforts to do a more intelligent, weighted fusion by estimating confidences were unsuccessful so far.

the whole utterance as a target for each syllable. The following features are used:

- The general purpose prosodic features as described in Section III-B1 for the current syllable, for its nucleus, and for the word the syllable belongs to, contexts '$-2,-2$', '$-2,-1$', ..., '$+2,+2$' (312 features),
- mostly binary features encoding primary/secondary word accent and prototypical phrase boundaries in the neighbourhood of ± 2 syllables, position of the current syllable witin the word and utterance (60 features),
- features encoding prototypical phrase accent, number of syllables, position in utterance etc. of the word the current syllable belongs to (10 features), and
- the number of words, and the sentence mood (statement, exclamation, question) of the utterance (four features).

Again, we tried to capture all potentially relevant information, accepting high redundancy within the feature set, and leaving it to machine learning algorithms to find out the actually relevant ones. We hope that this new divide-and-conquer may:

- Possibly provide a higher robustness because the SVR is trained with more instances and less features than in global model, and the final utterance score is an average over many single scores;
- capture information that is lost when levelling down the utterance to the global features as described in Section III-B2, by the precise, chronological context of a syllable represented by the features, and

A weak spot is definitely the use of utterance level scores. We tried to obtain syllable scores by a bootstrapping approach; we got promising but no conclusive results yet.

An obvious extension of the approach is to enrich each syllable's features with the global utterance features *Pros* or *All*. Accordingly, the local models will be referenced as *Local*, *Local+Pros* and *Local+All* in the evaluation.

V. Experiments and Results

A. Nuisance Removal

As discussed, obtaining representative training/testing data is difficult. Sometimes, however, there may be prior knowledge that might help to make the data more representative, or the evaluation more realistically, e. g. a known invariance that can be taken into account. When rating rhythm, such an invariance might be the speaking rate: Ideally, pronunciation scores should be invariant against tempo (within reasonable limits); after all, native speakers can speak fast or slow and should get good scores always. On the other hand, good learners (with good pronunciation scores) tend to speak faster than poor learners (with worse pronunciation scores). Thus, tempo is definitely a useful feature for automatic scoring. So when building a system for application, one will want to utilize tempo, but for judging the aptness of features, it can be interesting to study what happens when ignoring it. In order to do so, we estimate tempo by the average syllable duration T, and alter the *reference* Y such that it is no more correlated to T:

$$Y' = Y - \frac{\mathrm{Cov}(Y,T)}{\mathrm{Var}(T)} \cdot T. \tag{4}$$

For the C-AuDiT data, the syllable duration correlated indeed strongly (0.59) with the *pros* rating, which might partly be explained by the reading style or reading-related difficulties of the learners. Removing the correlation to duration from the labels did not affect the reliability much: the expected correlation of the combined labels to the ground truth fell from 0.99 to 0.98. For dod, duration is not correlated so strongly with *pros* (0.23), but the expected correlation of the combined labels to the ground truth suffered a bit because of fewer available labellers (five): it dropped from 0.85 to 0.80.

B. Choice of Meta-Parameters

In order to get useful results with our model, it is vital to choose suitable meta-parameters for SVR. All input features are transformed to lie within $[0;1]$ as commonly done. As kernel functions, we considered only the linear kernel and the normalized polynomial kernel [28] with lower orders, i. e.

$$\tilde{\mathrm{K}}(\boldsymbol{x},\boldsymbol{y}) = \frac{\mathrm{K}(\boldsymbol{x},\boldsymbol{y})}{\sqrt{\mathrm{K}(\boldsymbol{x},\boldsymbol{x})\mathrm{K}(\boldsymbol{y},\boldsymbol{y})}} \tag{5}$$

$$\text{with} \quad \mathrm{K}(\boldsymbol{x},\boldsymbol{y}) = (\langle \boldsymbol{x},\boldsymbol{y}\rangle + 1)^p \tag{6}$$

for exponent $p \in \{4, 8\}$. In our experience, this normalization of the vectors in feature space tends to improve performance, shorten training times and facilitate the optimization of the complexity parameter C. We try all combinations of $C \in \{0.001, 0.01, 0.1, 1\}$ and the three kernels and only report the best result. Strictly speaking, we would have to optimize these meta-parameters automatically inside a cross-validation loop, but given the need to evaluate speaker- and utterance-independently (see below), this would entail a 4-fold *nested* cross-validation, i. e. prohibitive computational costs. Thus, we effectively optimize the meta-parameters on the test set, but the effect of overfitting should be small since we are only optimizing two parameters, and only very coarsely.

C. Speaker-Independent Evaluation

In a first set of experiments, we calculate the accuracy of different systems in a two-fold *speaker-independent* cross-validation, i. e. in each of the two folds, half of the data is used for training, and the other half – disjunct w. r. t. speakers – is used for testing.

D. Speaker- and Utterance-Independent Evaluation

In a second set of experiments, we evaluate *speaker- and utterance-independently*, i. e. each pair of training and test set has to be disjunct with respect to speakers *and* utterances. For C-AuDiT, we perform a five-fold utterance (i. e. leave-one-out) cross-validation within a two-fold speaker cross-validation, i. e. for each of the $2 \times 5 = 10$ folds, $4/5 \times 1/2 = 40\%$ of the data can be used for training, $1/5 \times 1/2 = 10\%$ for testing, and 50% cannot be used. For dod, we perform just a two-fold utterance cross-validation within a two-fold speaker cross-validation, i. e. for each of the $2 \times 2 = 4$ folds, $1/2 \times 1/2 = 25\%$

of the data can be used for training, the same amount for testing, and 50% cannot be used.

E. Results

Results for different feature sets and models are given in Table I.

1) C-AuDiT: The first row (not counting headlines) of Table I refers to the most optimistic evaluation criteria: unchanged reference scores, and just speaker-independent evaluation, i. e. estimated performance for an assessment task on previously known sentences. Here, all feature sets and models, apart from *GPI* already yield relatively high correlations $\geq$ 0.58. *Rhy-All* (0.73) already scores almost the maximally reached correlation (0.75: *Pros*, *All*, *Local+Pros*, *Local+All*).

We now study how the feature sets perform when the plainest feature, the average syllable duration, is excluded from competition by making the reference uncorrelated to it, see the second row in Table I. For the *Iso* features (0.22) we now see that the previous success (0.58) was largely owed to also coding duration. The other rhythm features *GPI* (0.34) and especially *PVI* (0.45) are more successful in coding rhythm quality beyond tempo. The combination, *Rhy-All* yields no less than 0.54. The 'brute-force' approach *Pros* now is a bit clearer ahead of its 'expert-tailored' competitor *Rhy-All* with 0.58, and *Local+Pros* and *Local+All* are in the lead by a whisker with 0.59.

When evaluating also sentence-independently, results generally drop quite dramatically (see third and fourth row of Table I). When allowing the use of duration (third row), i. e. estimate the performance for an assessment task on arbitrary sentences, *Rhy-All* now only scores 0.58 (vs. 0.73 in sentence-dependent evaluation), and *Pros* and *All* drop even further to 0.53 (vs. 0.75). Apparently, the high number of features compared to the number of training instances ($50\% \times 5 * 94 = 235$) presents a problem here for generalizing to unknown sentences, otherwise *All* (which includes *Rhy-All*) would not score worse (0.53) than *Rhy-All* and *Local+Pros* and *Local+All* (0.54) would not be worse than *Local* (0.64). This bring us to the best performing approach in this setting, *Local*, which scores still 0.64 in this setting. Obviously, here the efforts for more robustness IV-B bear fruit. Combining *Pros* and *Local* by late fusion (not contained in Table I; weights – very coarsely – optimized on test: 0.3 resp. 0.7), we can further improve the correlation to 0.67.

Looking at the modelling power beyond duration (fourth row), *Rhy-All* is almost at the top (0.33), clearly beating *Pros* (0.26), presumably again caused by too few training instances to take advantage of the feature set. At least, combining the local with the global approach (*Local+Pros*) catches up (0.33 too), and *Local+All* just manages to be in the lead with 0.34.

2) dod: Here, results show a similar pattern, but sentence-dependent performance (fifth and sixth row) is only a bit better than sentence-independent performance (seventh and last row) is much less pronounced, which is due to the fact that *dod* contains much more different sentences (410 vs. 5). This reduces the danger of overfitting to the sentences (or the ability to adapt, for text-dependent tasks). Also, the difference between original (rows five and seven) and duration-deprived reference (sixth and last rows) is less clear, since we have seen that duration is only correlated to *pros* with 0.23 on *dod*.

The relevant results for a sentence-dependent assessment task (row five) show just a very slight preference for *Pros* (0.57) when compared to *Rhy-All* (0.56). Also for a sentence-independent assessment task (row seven), *Pros* (0.52) is only a little ahead of *Rhy-All* (0.49). The divide-and-conquer approach was less successful: *Local* did not score more than 0.48 here. Late fusion of *Pros* with *Local* improved the result by a fraction to 0.53 (not contained in Table I; weights: 0.3 resp. 0.7).

Regarding modelling power beyond duration, *Pros* (0.47) could again be shown to be noticeably better than *Rhy-All* (0.42). As for *C-AuDiT*, *Local+All* just manages to be in the lead with 0.48. For this most difficult task, the best results are clearly ahead of those of *C-AuDiT* (maximally 0.34, see *Local+All* in row four), which is owed to the better representativeness of *dod*.

F. Discussion

1) Language Testing: The sentence-dependent results (up to 0.75) for C-AuDiT are quite good; however, the applicability for CAPT is limited. For language *testing*, however, where only a finite set of test items is needed, it is perfectly feasible only to use sentences already contained in the training set of an automatic scoring method. Nevertheless, it is questionable whether the discriminative approach pursued is best suited for this task. After all, the needed data collection is very costly, and some adaptations (calibration/partitioning of the material) had to be employed because the method cannot adapt to too many sentences at once (see the performance drop for *dod*). However, as test items can be defined *a priori*, a generative approach based on native templates along the lines of [29] is much cheaper and may yield similar or even superior results: the number of needed test items is – compared to a CAPT application – much smaller, so one can afford to record template utterances by many speakers, presumably leading to meaningful distance measures.

2) CAPT: The best sentence-independent result for *C-AuDiT* was a correlation of 0.67 to the reference. Taking into account the quality of the reference using Equation 3, this means that the system has an expected correlation of $0.67 \cdot 0.99 = 0.66$ to the ground truth of *pros*. This is clearly better than the performance of the average individual labeller (0.54) probably owed to the fact that *C-AuDiT* is a relatively easy task due to the reading prosody/reading difficulties, which allows the automatic system, naturally adapting to the given domain, to take a 'shortcut' for rating prosody. For *dod*, the corresponding best automatic result was 0.53. Given the relatively low quality of the reference, this means that the system has a correlation of $0.53 \cdot 0.85 = 0.45$ to the ground truth. The average labeller on the other hand is clearly ahead with 0.58. Our interpretation is that *dod* is the more difficult task because reading difficulties play a smaller role, and the

TABLE I
RESULTS FOR DIFFERENT FEATURE SETS AND MODELS FOR C-AUDIT (UPPER HALF) AND DOD (LOWER HALF) IN TERMS OF PEARSON CORRELATION COEFFICENT BETWEEN THE SYSTEM'S OUTPUT AND THE REFERENCE. 'SPEAKER' STANDS FOR A SPEAKER-INDEPENDENT EVALUATION, 'SPEAKER+SENTENCE' FOR A SPEAKER- AND SENTENCE-INDEPENDENT EVALUATION. 'ORIG' REFERS TO TAKING THE ORIGINAL COMBINED RATINGS OF ALL LABELLERS AS A REFERENCE; FOR 'W/O DUR' THE CORRELATION TO THE AVERAGE SYLLABLE DURATION HAS BEEN REMOVED.

Corpus	Evaluation	Reference	*Dur*	*Iso*	*PVI*	*GPI*	*Rhy-All*	*Pros*	*All*	*Local*	*Local+Pros*	*Local+All*
C-AuDiT	Speaker	Orig	0.60	0.58	0.59	0.45	0.73	**0.75**	**0.75**	0.69	**0.75**	**0.75**
		w/o Dur	0.14	0.22	0.45	0.34	0.54	0.58	0.58	0.50	**0.59**	**0.59**
	Speaker+Sentence	Orig	0.54	0.50	0.42	0.19	0.58	0.53	0.53	**0.64**	0.54	0.54
		w/o Dur	-0.13	-0.15	0.25	0.16	0.33	0.26	0.28	0.21	0.33	**0.34**
dod	Speaker	Orig	0.48	0.50	0.41	0.37	0.56	**0.57**	**0.57**	0.53	**0.57**	**0.57**
		w/o Dur	0.40	0.43	0.38	0.34	0.50	0.52	0.52	0.50	**0.53**	**0.53**
	Speaker+Sentence	Orig	0.40	0.45	0.33	0.31	0.49	**0.52**	**0.52**	0.48	0.51	**0.52**
		w/o Dur	0.32	0.37	0.31	0.28	0.42	0.47	0.47	0.44	0.47	**0.48**

speech sounds more spontaneous. Thus, with the 'crutches' no longer available, the automatic systems are revealed to still perform with sub-human performance. However, we take some comfort in the conviction that as soon as we hire some more labellers, we will get somewhere near the performance of the average human: the quality of the reference will be increased by more labellers, and it is also likely that the system will show a higher correlation to better labels. Thus, we can expect to increase both factors of Equation 3.

VI. OUTLOOK

A. More Features

Promising approaches to feature extraction for the discriminative approach are, e. g., the GMM-UBM super-vector approaches [30] and prosodic contour features [31] developed in the field of speaker identification, or the combination of both approaches [32], [33].

B. Generative Approach

It remains to be answered whether the complexity of rhythm is not too high for the chosen discriminative approach, given the efforts required for collecting suitable data. Possibilities for generative approaches would be a counterpart to the GOP algorithm for discrete prosodic events such as boundaries and accents which can be recognized or decoded with reasonable accuracy [25], or using conditional densities to make do without discrete prosodic classes.

C. Feedback

Up to now we have concentrated only on assessment and have completely ignored feedback. It would be interesting to study whether approaches based on machine learning can contribute to giving useful feedback. An example could be assessment modules that only use specific prosodic aspects such as loudness or duration to derive more specific feedback on what the learner should concentrate on in order to improve. Another improvement would be more localized feedback; possibly, the *Local* approach could be extended by bootstrapping to derive and predict syllable-level scores.

VII. CONCLUSION

The impact of suboptimal non-native prosody on understanding is well-known and has received some attention lately. In this article, we wanted to contribute to some of the most basic questions related to this topic; to this aim, we collected and annotated two databases with English as L2, spoken by speakers of different L1. The data were (1) read or prompted. We implemented (2) specialized rhythm features suggested in the phonetic literature as well as (3) a large feature set comprising general-purpose prosodic features. We addressed (4) the differences in employing different types of more or less expert labellers and (5) different numbers of labellers, and computed, based on comparing the labeller, (6) estimates of the effective quality of averaged annotations and automatic scores. Evaluation was done (7) speaker-independently and (8) utterance-independently. We showed the relevance of steps (1) to (8), based on correlations obtained for regression models with the human reference. Eventually, we discussed the impact of the single steps on performance and usability in real-life CAPT application.

ACKNOWLEDGMENTS

This work was funded by the German Federal Ministry of Education, Science, Research and Technology (*BMBF*) in the project *C-AuDiT* under Grant 01IS07014B, and by the German Ministry of Economics (*BMWi*) in the project *AUWL* under grant KF2027104ED0. The responsibility lies with the authors. The perception experiments were conducted/supervised by Susanne Burger (Pittsburgh), Catherine Dickie and Christina Schmidt (Edinburgh), and Tanja Ellbogen and Susanne Waltl (Munich). We want to thank Andreas Maier for adapting PEAKS to our task.

REFERENCES

[1] M. Piat, D. Fohr, and I. Illina, "Foreign accent identification based on prosodic parameters," in *Proc. Interspeech*, Brisbane, Australia, 2008, pp. 759–762.

[2] J. Tepperman and S. Narayanan, "Better nonnative intonation scores through prosodic theory," in *Proc. Interspeech*, Brisbane, Australia, 2008, pp. 1813–1816.

[3] J. Lopes, I. Trancoso, and A. Abad, "A nativeness classifier for TED talks," in *Proc. ICASSP, Prague, Czech Republic*, 2011, pp. 5672–5675.

[4] F. Hönig, A. Batliner, K. Weilhammer, and E. Nöth, "Islands of failure: Employing word accent information for pronunciation quality assessment of english L2 learners," in *Proceedings of SLATE*, Wroxall Abbey, 2009, no pagination.

[5] E. Grabe and E. L. Low, "Durational variability in speech and the rhythm class hypothesis," in *Laboratory Phonology VII*, C. Gussenhoven and N. Warner, Eds. Berlin: de Gruyter, 2002, pp. 515–546.

[6] F. Ramus, "Acoustic correlates of linguistic rhythm: Perspectives," in *Proc. Speech Prosody*, Aix-en-Provence, 2002, pp. 115–120.

[7] D. Abercrombie, *Elements of General Phonetics*. Edinburgh: University Press, 1967.

[8] H. Niemann, *Klassifikation von Mustern, 2. Auflage*. Heidelberg: Springer, 2003.

[9] B. Schuller, S. Steidl, and A. Batliner, "The INTERSPEECH 2009 emotion challenge," in *Proc. Interspeech*, Brighton, 2009, pp. 312–315.

[10] H. Günther, "Zur methodischen und theoretischen Notwendigkeit zweifacher statistischer Analyse sprachpsychologischer Experimente. Mit einer Anmerkung von R. Kluwe. / methodological and theoretical arguments for two-fold statistical analysis in psycholinguistic experiments. with comments by R. Kluwe," *Sprache & Kognition*, vol. 3, no. 4, pp. 279–285, 1983.

[11] S. M. Witt, "Use of speech recognition in computer-assisted language learning," Ph.D. dissertation, Univ. of Cambridge, 1999.

[12] F. Hönig, A. Batliner, K. Weilhammer, and E. Nöth, "How many labellers? Modelling inter-labeller agreement and system performance for the automatic assessment of non-native prosody," in *Proc. SLATE*, Tokyo, Japan, 2010, no pagination.

[13] R. Snow, B. O'Connor, D. Jurafsky, and A. Y. Ng, "Cheap and fast - but is it good? evaluating non-expert annotations for natural language tasks," in *Proc. of the Conference on EMNLP*, Honolulu, Hawaii, 2008, pp. 254–263.

[14] W.-H. Lin and A. Hauptmann, "Vox populi annotation: Measuring intensity of ideological perspectives by aggregating group judgments," in *Proc. LREC*, Marrakesh, 2008.

[15] F. Hönig, A. Batliner, and E. Nöth, "How many labellers revisited – naïves, experts and real experts," in *Proc. SLATE*, Venice, Italy, 2011, no pagination.

[16] F. Hönig, A. Batliner, K. Weilhammer, and E. Nöth, "Automatic assessment of non-native prosody for English as L2," in *Proc. Speech Prosody*, Chicago, 2010, no pagination.

[17] M. Grimm and K. Kroschel, "Evaluation of natural emotions using self assessment manikins," in *Proc. ASRU*, San Juan, Puerto Rico, 2005, pp. 381–385.

[18] V. C. Raykar, S. Yu, L. H. Zhao, G. H. Valadez, C. Florin, L. Bogoni, and L. Moy, "Learning from crowds," *The Journal of Machine Learning Research*, vol. 11, pp. 1297–1322, 2010.

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

[20] A. Batliner, R. Kompe, A. Kießling, M. Mast, H. Niemann, and E. Nöth, "M = Syntax + Prosody: A syntactic–prosodic labelling scheme for large spontaneous speech databases," *Speech Communication*, vol. 25, pp. 193–222, 1998.

[21] A. Maier, T. Haderlein, U. Eysholdt, F. Rosanowski, A. Batliner, M. Schuster, and E. Nöth, "PEAKS - a system for the automatic evaluation of voice and speech disorders," *Speech Communication*, vol. 51, pp. 425–437, 2009.

[22] Council of Europe, Ed., *Common European Framework of Reference for Languages: Learning, Teaching, Assessment*. Cambridge University Press, 2001, available as PDF from www.coe.int/portfolio, last visited 11th April 2012.

[23] P. M. Bertinetto and C. Bertini, "On modeling the rhythm of natural languages," in *Speech Prosody 2008, May 6-9, 2008, Campinas, Brazil*, 2008.

[24] V. Dellwo, "Influences of speech rate on the acoustic correlates of speech rhythm. an experimental phonetic study based on acoustic and perceptual evidence," Ph.D. dissertation, Rheinische Friedrich-Wilhelms-Universität Bonn, 2010.

[25] A. Batliner, J. Buckow, H. Niemann, E. Nöth, and V. Warnke, "The Prosody Module," in *Verbmobil: Foundations of Speech-to-Speech Translation*, W. Wahlster, Ed. Berlin: Springer, 2000, pp. 106–121.

[26] A. Batliner, S. Steidl, C. Hacker, E. Nöth, and H. Niemann, "Tales of tuning – prototyping for automatic classification of emotional user states," in *Proc. Interspeech*, Lisbon, Portugal, 2005, pp. 489–492.

[27] M. Hall, E. Frank, G. Holmes, B. Pfahringer, P. Reutemann, and I. H. Witten, "The WEKA data mining software: An update," *SIGKDD Explor. Newsl.*, vol. 11, pp. 10–18, November 2009.

[28] A. B. A. Graf, A. J. Smola, and S. Borer, "Classification in a normalized feature space using support vector machines," *IEEE Transactions on Neural Networks*, vol. 14, no. 3, pp. 597–605, 2003.

[29] J. Tepperman, T. Stanley, K. Hacioglu, and B. Pellom, "Testing suprasegmental english through parroting," in *Proc. Speech Prosody*, Chicago, 2010, no pagination.

[30] W. Campbell, D. Sturim, and D. Reynolds, "Support vector machines using GMM supervectors for speaker verification," *Signal Processing Letters, IEEE*, vol. 13, pp. 308–311, 2006.

[31] M. Kockmann, L. Burget, and J. Černocký, "Investigations into prosodic syllable contour features for speaker recognition," in *Proc. ICASSP*. IEEE Signal Processing Society, 2010, pp. 4418–4421.

[32] N. Dehak, P. Dumouchel, and P. Kenny, "Modeling prosodic features with joint factor analysis for speaker verification," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 15, no. 7, pp. 2095–2103, 2007.

[33] M. Kockmann, "Subspace modeling of prosodic features for speaker verification," Ph.D. dissertation, Brno University of Technology, Faculty of Information Technology, 2012, to appear.

Mining pronunciation data for Consonant cluster problems

Garrett Pelton
Carnegie Speech Company
925 Liberty Ave. Pittsburgh PA 15222 USA
gap@carnegiespeech.com

Abstract— **This paper describes mining data collected from users of NativeAccent™, to decide if these users had pronunciation problems within consonant clusters that warranted special exercises and lessons. Consonant clusters are a known issue for English learners, partly because many other languages either don't have consonant clusters or because they have different sets of clusters. We look at 3000 NativeAccent users. These users' background includes eight different native languages. We look for evidence that consonant cluster problems are being detected, and for patterns that depend upon the user's native language. We find evidence that the users are seeing cluster problems and that some of the problems are the same across the 8 languages and some problems are more prominent in a few of the languages. We also discuss the implications of this data for the intelligent tutoring system within NativeAccent.**

Index Terms: pronunciation, error detection, tutoring system, assessment

I. INTRODUCTION

The past decade has seen improvements in the algorithmic detection of errors in non-native pronunciation [1] and [2]. This work has led to the development of products that can be used to improve a user's pronunciation. The work of Yamada [3] has been converted to a complete pronunciation training system for Japanese speakers who want to speak English. It has been commercialized by ATR in Japan. NativeAccentTM by Carnegie Speech is another commercial product used to train non-native speakers to improve their pronunciation of English.

NativeAccent started as the Fluency project [1] at Carnegie Mellon University. It has been converted to a complete pronunciation training system, and over the last 10 years thousands of people have used NativeAccent to improve their pronunciation. NativeAccent compares a user's pronunciation to a statistical model of native speakers. A close match is considered good pronunciation. We have these models for both Midwest American English speakers and British English from speakers whose speech is close to what used to be called BBC English. Rather than doing a detailed analysis of how NativeAccent detects pronunciation issues, this paper documents one of our explorations in how NativeAccent can be improved to give our users a better experience and to better address their specific problems. This exploration is done by examining the logs of user data looking for patterns of problems. The discussion section covers whether the detected user problems could be handled better by NativeAccent.

Pronunciation problems within consonant clusters are the focus of the problems investigated in this paper. Consonant clusters are where the native pronunciation of a word involves a sequence of consonants without intervening vowels. Consonant clusters are a known pronunciation problem in non-native speech [4]. For that reason, NativeAccent's curriculum includes consonant clusters exercises in its lessons. For example, the **/t/** lesson might start with exercises on simple words like "tap", "bat" and "later" but the lesson can also include words with "t" in a consonant cluster like "string". The question asked in this paper is whether we should change our detection algorithms and/or our tutoring because our users are exhibiting problems that our current system doesn't handle. This paper uses data mining on the logs of NativeAccent users to answer both this question and to provide more insight into the various consonant cluster problems our users exhibit.

II. BACKGROUND

NativeAccent's pronunciation system tutors students in the pronunciation of a target language. To do so we needed to develop: leveled corrective feedback information; a full curriculum; a student model; a strategy on how to proceed through the curriculum for different learners (fast and slow, for example); a reporting mechanism for the teacher (to follow individual and grouped student progress). NativeAccent has been endowed with these features as well as others, requested by the customers that are less essential to the basic system. Finally NativeAccent also includes non-

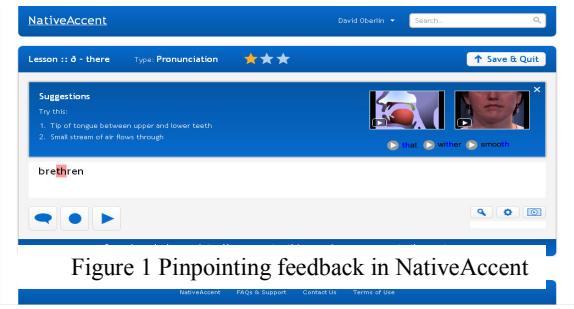


Figure 1 Pinpointing feedback in NativeAccent

pronunciation training components (grammar, word stress and fluency) that will not be described in this paper.

NativeAccent relies on a detection system for correct pronunciation called "pinpointing". Pinpointing results can be seen in Figure 1 on a **/ð/** (voiced "TH") sound within the **/ðr/** consonant cluster in the word "brethren. The screenshot in Figure 1 shows that the student mispronounced the **/ð/** sound because the graphemes (letters) associated with that sound are marked in red. If the student had pronounced the sound correctly, these same graphemes would be marked in green. NativeAccent keeps the user focused on one skill at a time, and since the user is in the **/ð/** lesson only the TH is marked in red for the user. Though pinpointing detects pronunciation problems in all the phones, it only shows the pinpointing results on the focus phone for that lesson. However, all the pinpointing results go into a student model that keeps track of the student's problem areas, and are reported in the logs.

Also in Figure 1, you can see some of the learning tools available to the student when an error is detected. The student can play a model speaker, and they can listen to themselves. It could be argued that self-discovery would be an alternative to the pinpointing presentation and valid method of learning here ("listen to what you said and find your own errors"). Yet without being trained on how to listen discriminatively, the task is extremely difficult and errorful for the student. Pinpointing provides an independent assessment of the speech. In addition, above the pinpointing display in Figure 1, there are both textual suggestions, and little movies showing both a frontal view of someone saying the sound, the articulators inside the mouth when producing the sound correctly. One issue for this paper is that all these tools are phonetic based, and we want to know if they are adequate in the case of consonant clusters.

Consonant clusters are sequences of consonants in the pronunciation of a word. The errors an ESL student makes within a consonant cluster can include the types of errors we see on single consonants like substitution, voicing, de-voicing, affrication, etc.. However, other errors are more typical of consonant clusters. For example, one of Carnegie Speech's native Mandarin speaking employees pronounces sphinx as /səfInəks/ rather than /sfinks/. After listening to some of the speakers pronouncing consonant clusters found by this study, we found similar problems in consonant clusters as are found single consonant contexts. However, we also found epenthesis issues like the error exhibited by the Carnegie Speech employee. In addition, as we shall see, we found a higher rate of errors in consonant clusters than we see in singleton consonants for some users.

The NativeAccent curriculum has about 800 exercises that cover all of the sounds in English as well as aspects of duration and pitch. Some of the exercises are specific to one or more of the 28 native languages (for example, the /TH/ exercises are not the same for Japanese learners and for Russians). The completeness of the curriculum allows us to select user group-specific exercise subsets. In addition, NativeAccent uses intelligent tutor [5] technology to focus the training on the individual needs of the student. By providing a smaller, very focused subset of lessons and exercises the students can improve their pronunciation within the amount of time they can afford to devote.

The learning improvement from the use of intelligent tutoring has been shown in many domains. In particular Anderson [5] shows reduction of 1/3 the training time to achieve a particular level of performance in learning a programming language. Koedinger et al in [6] shows an improvement of 1 standard deviation for students using an Algebra 1 intelligent tutor compared to similar students in a teacher run Algebra 1 class. Carnegie Speech has published studies [7] about how well our users do using this intelligent tutor approach. A more recent unpublished study of 120 people showed the 60 people who only participated in a teacher run pronunciation class (control group) improved 62% in 7 hours, and the 60 people in the test group using NativeAccent during ½ of their pronunciation class time improved 104%. This difference was not quite statistically significant (r=0.07), but the time using NativeAccent was relatively small. We see substantial improvement in about 10 hour of use.

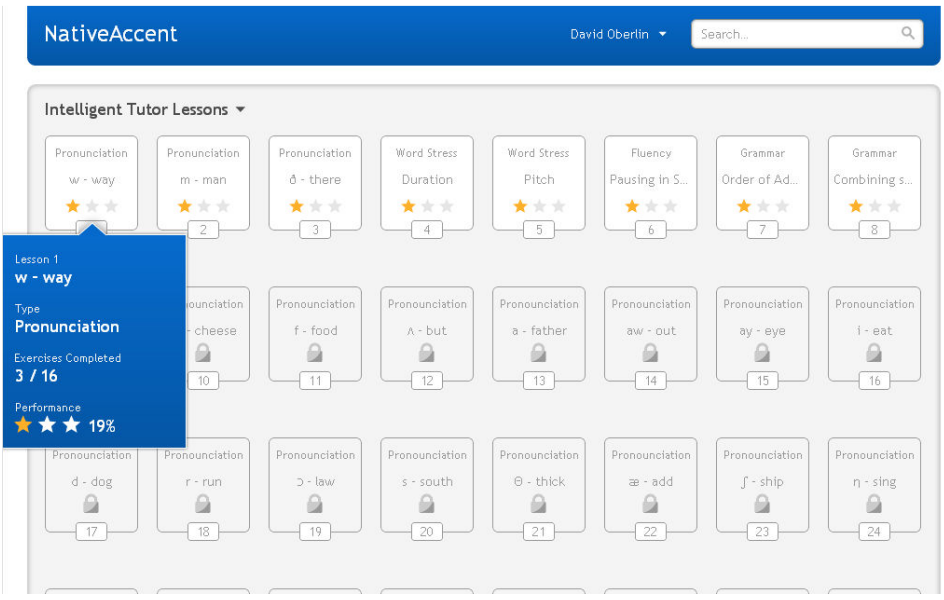


Figure 2*: Intelligent Tutor derived Lesson Plan*

Intelligent tutoring shows up in NativeAccent in two ways. The first is shown in Figure 2. The screenshot in Figure 2, shows a sequence of phone, stress, grammar, and fluency lessons. The user then simply follows the sequence. The Intelligent Tutor creates the sequence from the student's performance on a previously taken assessment. The assessment diagnoses the student's need and the lessons that best address the student's needs make up the sequence. The tutor picks enough lessons to fill a 6.5 hour lesson plan. This lesson plan is then repeated twice, to give a spaced repetition to each lesson topic. When a student enters a lesson, the intelligent tutor is invoked a second time to determine what exercises to give the student. For pronunciation, upon entering the lesson the first time, the student might get mostly short easy words. If the student does well with the short word exercises, upon their next entry they will be given harder words (perhaps with consonant clusters) and perhaps phrases and sentences. The intelligent tutor focusing of the student on their needs and ignoring the portions of the curriculum that the student doesn't need is why the student progresses so quickly.

Perfect intelligent tutoring relies upon knowing what skill deficiency is the root cause of a mistake and having curriculum that helps the student not make that mistake. The graphs showing the transformation of errors into learning curves once errors can be assigned to a cause (rule) in Anderson [5] are very compelling. The empirical results on tutors that can assign root causes supports that research. Pinpointing provides probabilistic root cause information for substitution, affrication, deletion and other mistakes related to a single phone. Co-articulation problems and those problems associated with a phone in a context, along with errors in automatic speech analysis reduce the effectiveness of the tutoring. However, these problems aren't systematic, and pinpointing does pick up on the systematic issues the user exhibits. Carnegie Speech's experience shows that the small amount of random error doesn't affect the tutoring overmuch.

The reason for looking at consonant clusters is they are a known problem, and we want to know if we need to do more to detect problems unique to consonant clusters and create special remediation for those problems. Alternatively we could treat consonant cluster problems as unique sources of errors, and then use the intelligent tutor to focus the user on errors within consonant clusters independently of the user's errors on singleton consonants. Celce-Murcia [8] shows examples of consonant cluster problems for English learners with different L1. Celce-Murcia also shows that the type of consonant cluster errors is at least partially dependent on the L1. Some of the problems she mentions (substitution, epenthesis, affrication, deletion etc.) we also have seen in consonant clusters. However, some of these problems are not unique to consonant clusters, and our users reduce these errors in general. Does the seen improvement in the user's phoneme production mean that we don't have to provide any specific support for the user's production of consonant clusters? This paper is investigating whether there seems to be systematic errors in consonant cluster production that aren't handled by our current system.

Table 1:Number of users by Native Language in dataset

Native Language	# users	Native Language	# users
Arabic	54	Malayalam	58
Armenian	4	Mandarin	820
Bahasa Ind	6	Marathi	23
Bengali	38	Mongolian	6
Burmese	1	Nepali	1
Cantonese	177	Other	255
Cebuano	3	Polish	18
Chinese	5	Portuguese	26
Croatian	1	Punjabi	30
English	77	Romanian	50
Farsi	15	Russian	352
French	66	Serbian	2
German	17	Slovak	11
Greek	3	Spanish	1130
Haitian Cr	5	Tagalog	355
Hebrew	9	Taiwanese	4
Hindi	198	Tamil	168
Italian	4	Telugu	252
Japanese	124	Thai	96
Kannada	18	Turkish	16
Korean	284	Ukrainian	22
Kyrgyz	1	Urdu	28
Lao	1	Vietnamese	218
		Grand Total	5052

III. DESCRIPTION OF THE DATA

The data consists of a set of logs from 6000 users of NativeAccent. The users used NativeAccent for varying amounts of time. We removed the users that used the system longer than 365 days or less than 5 days. The users with more than 365 days were internal Carnegie Speech users demonstrating the software or testing it. The people with less then 5 days use were assumed to be people trying out the software and then deciding not to use it. We also eliminated the first 200 users which again were mostly assumed to be testing the software. This left us with 5052 users of NativeAccent.

These users were spread over 44 different languages as shown in Table 1. The "Other" native language category in Table 1 is selected when the user can't find their native language in the 60 languages in NativeAccent, and doesn't know which language on the list is similar. Eliminating all native languages with less than 175 users, and the Other category users, leaves us with 3786 users, and just the 8

Native Language	# users	/b/	/tʃ/	/d/	/ð/	/f/	/g/	/h/	/dʒ/	/k/	/l/	/m/	/n/	/ŋ/	/p/	/r/	/s/	/ʃ/	/t/	/θ/	/v/
Cantonese	48	2%			8%		5%	2%	2%	1%	8%	2%	1%	4%	5%		1%	5%	2%		9%
Hindi	67	11%			11%		3%		2%	3%	8%	11%		8%				5%		3%	1%
Korean	79	3%	1%		7%	1%	9%	1%	2%		4%	3%		5%	2%	1%		5%	3%		3%
Mandarin	230	3%			7%		7%	1%	3%		15%	3%		6%				3%		1%	5%
Russian	140	1%			18%		1%				1%	4%		20%	1%			10%	3%		9%
Spanish	438	11%		3%	6%		4%		1%	2%	4%	6%		8%	1%			5%	3%	6%	3%
Telegu	62	6%			13%		4%	1%	1%	3%	9%	6%		9%		2%		4%		2%	7%
Vietnamese	50	1%	2%	3%	6%		4%		4%	1%	5%	3%	2%	8%	2%			5%	10%	1%	4%

Table 2: Percentage of users who had **MORE** difficulty with a phoneme within a consonant cluster

languages of Cantonese, Hindi, Korean, Mandarin, Russian, Spanish Telugu and Vietnamese.

These users came mainly from the college and community colleges of the US. In addition, some came from work force development organizations in Florida. A reasonable deduction would be that these users were in their late teens to 25, though Carnegie Speech doesn't know. Likewise most schools don't treat pronunciation as an early English learner problem, so it is likely that these students were at an NRS level 3 (about a year of English training) or better, though again Carnegie speech doesn't know.

These students used NativeAccent for varying amounts of time, from 8 minutes to 54 hours. Not all this time was spent on pronunciation exercises. A pronunciation exercise has the student recording themselves reading a word, phrase, or sentence out loud (as seen in Figure 1). Every time a student records their voice in a pronunciation lesson a log file is created. This log file contains an assessment of how well the student spoke every phoneme of the sentence they were asked to read. These log files form the basis of the data that was analyzed.

IV. RESULTS

Our goal is to determine if users of NativeAccent are having significantly more problems saying a consonant within a cluster compared to outside of a cluster. Outside of a cluster means that the consonant was either surrounded by vowels, or occurred at the beginning or end of a word, and the adjacent phoneme in the word is a vowel. We didn't analyze the data for inter-word consonant cluster effects.

We separated all the instances of consonant production for each user into two groups: within a consonant cluster and outside of a consonant cluster. We had separate groups for each consonant. Significantly more problems means that the distribution of errors for the consonant within a cluster had minimal overlap with the distribution of errors for the consonant outside of a cluster. Each user performance of a consonant has either a success or failure outcome, thus the set of outcomes for that user on each consonant is best modeled with a binomial distribution. If the Wilson Score Interval (confidence level 95%) showed the consonant-within-cluster distribution didn't overlap at all with the consonant-outside-cluster distribution, we counted the two distributions for that consonant as being different.

We were looking for significant differences between a user's performance on consonants within a cluster and their performance outside a cluster. Only 3016 users had a significant difference in their performance.

The first analysis looked at the data from the phonetic point of view, counting the number of users who had significantly different error rates for that phone being within a cluster and that phone not being within a cluster. This analysis is shown in Table 2 and Table 3. Each row of Table 2 shows the percentage of native-language users that had significantly higher error rates on pronouncing each column's phone within a cluster compared to pronouncing it outside of a cluster. For instance 11% of the Spanish native language speakers had problems with the **/b/** phone within a cluster compared to outside a cluster. This effect might show up in words like "blue", "cabs", Table 2 doesn't distinguish between the different clusters. Table 3 is the converse. Each row of Table 3 shows the percentage of native-language users that had significantly lower error rates on pronouncing each column's phone within a cluster compared to pronouncing it outside of a cluster. The phones in Table 2 and Table 3 are the only phones where more than 1% of the speakers of any of our eight languages had a significant difference between the within cluster and outside cluster performance.

The existence of Table 3 where performance of the user is significantly better within a consonant cluster is somewhat

Native Language	/b/	/tʃ/	/d/	/ð/	/f/	/g/	/h/	/dʒ/	/k/	/l/	/m/	/n/	/ŋ/	/p/	/r/	/s/	/ʃ/	/t/	/θ/	/v/
Cantonese		4%	1%					2%									1%			
Hindi	1%	2%							1%					2%					1%	
Korean	2%	4%	2%					3%						2%		1%			2%	
Mandarin		4%						1%									3%		1%	
Russian								3%	1%					1%						
Spanish		2%				2%		5%												
Telegu	1%	4%				3%	2%	2%									2%			
Vietnamese						1%							1%		1%		1%			

Table 3:Percentage of users who had **LESS** difficulty with a phone within a consonant cluster

Native Language	/bz/ cabs	/fr/ frame	/fy/ fuel	/gz/ jogs	/kl/ likely	/ky/ cue	/ld/ old	/ly/ value	/ŋʃ/ anxious	/pl/ apply
Cantonese	5%						3%	5%		4%
Hindi	11%	3%	8%				8%	5%		
Korean	5%		4%	6%		4%		5%		
Mandarin	5%		4%	3%	4%			8%		5%
Russian	4%		9%					9%	3%	
Spanish	17%		5%				4%			
Telegu	6%		4%				8%	4%		
Vietnamese			4%	3%				3%		3%
Native Language	**/rd/ card**	**/rr/ harder**	**/rdʒ/ large**	**/rl/ yearly**	**/sk/ desk**	**/st/ mist**	**/str/ street**	**/ts/ nets**	**/vr/ average**	**/zd/ buzzed**
Cantonese			3%		3%				3%	
Hindi										
Korean							4%			
Mandarin										
Russian		3%	4%				5%		3%	
Spanish				4%			5%			4%
Telegu	3%									
Vietnamese						6%		4%		

Table 4: Percentage of users that had significantly more problems on particular clusters

surprising. The percentages are small, meaning a small number of users exhibited this behavior. Perhaps individual difficulties in pronouncing a consonant with a vowel account for these data.

The second analysis looked at the clusters where the users make significantly more errors on some phoneme in the cluster compared to the outside-of-cluster performance of the worst phone in the cluster. This analysis is shown in Table 4. Each cell of Table 4 shows the percentage of native language users that had significantly higher error rates on any phone in the cluster compared to pronouncing any of the phones in the cluster outside of clusters. For example, the **/fy/**cluster was problematic for every language but Vietnamese. This cluster occurs in the words "refuse" and "nephew". Nine percent of the Russian speakers had significantly more problems speaking the combination **/fy/** (either phone could be marked as bad) than with speaking either the **/f/** or **/y/** phone outside of a cluster like in "fine" or "you". In this second analysis, the threshold of including a cluster was that the cluster affected at least three percent of the users of any particular language. This higher threshold, compared to analysis 1, was chosen because there were a large number of clusters around the 1-2% that we didn't think useful to show.

There is no Table 5 that corresponds to Table 3. We found no clusters were the more than .5% of the users had done substantially better on the worst phone in the cluster than they had done on all of the phones in outside-cluster situations. We thought this level of problems was probably caused by individual differences rather than systemic problems, and wouldn't be significant.

V. Discussion of Results

This is the first analysis of a large corpus of spoken speech looking at consonant cluster problems. Altenberg [9] deals with 8 subjects all native Spanish speakers, and only looking at word initial problems. Similar studies with a small number of subjects have been done on other languages. The existence of pinpointing has enabled this analysis. At this time, we don't know the cause of most of the results, but the results themselves are interesting and hopefully can lead to better ESL training.

The first analysis shows that consonant clusters are not a major problem for all NativeAccent users. The number of users with "significant" cluster problems (as shown in Table 2 was 1/6th of the total number of users in our initial dataset. However, it is a problem for this small subset. We also wanted to know how uniform the cluster problems were

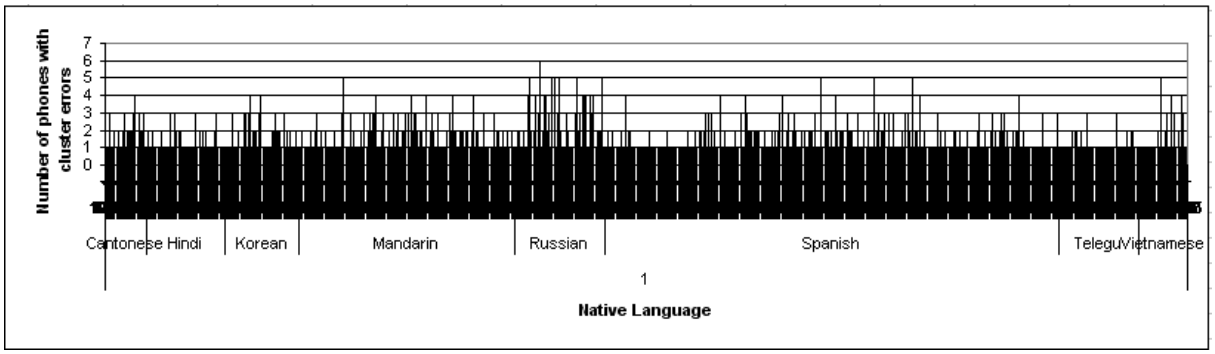


Figure 3: Number of phones with significant cluster errors per user

across our user base. Figure 3 shows the number of errors made per user. The conclusion we draw from Figure 3 is with our measurement technique, most of the users had one consonant cluster error. Only a small number of users had multiple errors. The multiple error users are the spikes in Figure 3. Native Russian speakers seem more likely to have multiple errors.

The results show there are only 13 phones that would seem to benefit from specific lessons and remediation. These are the phones where a number of languages showed a problem (>2%) in Table 2. This small number of problems was unexpected, as was the small number of clusters with problems in Table 2. Our guess is that our criterion for finding a consonant cluster is too strict. One way to weaken the criterion is to allow for a user some overlap of the distribution of consonant cluster problems with the distribution of outside of cluster problems. A further research item is to investigate what the right criteria is for diagnosing a problem as a consonant cluster problem.

The results show user's native language does make a difference. 10% of the Vietnamese speakers had problems the **/t/** phone where almost nobody else did. The second analysis would have us concentrate on **/ts/** and **/st/** clusters for these Vietnamese speakers. This is the type of information Carnegie speech wanted to get from this study. We have mechanisms within the intelligent tutor for creating initial biases based the native language. We could use this type of information to improve the Vietnamese experience.

VI. Conclusions

Pinpointing seems to be a blunt tool for looking at consonant cluster errors. If a user does a schwa insertion, then pinpointing will assigned the extra audio to one of the consonants on either side in the cluster. It is unclear which consonant will be marked wrong. Likewise Co-articulation affects could mask what actually happened. We need further investigations into either how to better use the current pinpointing, or in building better tools more focused on consonant cluster problems.

To determine better tools, we need to listen to a number of the recordings of students with known consonant cluster issues. Pronunciation transcription is a very time consuming and error prone task. We have listened to 78 recordings of the STR cluster and this shows that pinpointing was correct there was a problem, and the problems are mainly substitution, affrication, deletion and problems pronouncing the /r/ phone. These types of pronunciation problems are handled by the intelligent tutor. Epenthesis problems were less than 10% of the problems. It isn't clear from the data that specialized lessons are needed, compared to adding the ability of the intelligent tutor to focus the user's practice on consonant cluster problems when they appear.

We believe that the methodology used in this paper underestimates the real consonant cluster pronunciation problem rate. We are only declaring that a consonant cluster problem exists when the pronunciation of the phone outside of a consonant cluster is significantly better. We don't know if better detection would allow us to focus on a consonant cluster earlier, to a better effect.

What is clear is that we should be tracking consonant cluster issues better in our intelligent tutor, and focusing the student on the consonant cluster components of our current exercises when it is needed. In addition there are some low hanging fruit where we should be able to create more specific curriculum for some of the more pervasive problems, like the /bz/ cluster for Spanish speakers.

The way the intelligent tutor selects lessons and exercises, it could be worthwhile to create a single lesson that focused on consonant clusters. This lesson would only be selected if the user was demonstrating consonant cluster issues. Once this lesson is selected the intelligent tutor could select which exercises to give depending upon the clusters the student was having problems with.

VII. References

[1] M. Eskenazi, "Detection of foreign speakers' pronunciation errors for second language training - preliminary results," in *Proc. International Conference on Spoken Language Processing*, Philadelpia, 1996.

[2] L. Neumeyer, H. Franco, V. Digalakis, and M. Weintraub, "Automatic scoring of pronunciation quality," *Speech Communication*, vol. 30, no. 2, pp. 83–93, Feb. 2000.

[3] R. Akahane-Yamada, H. Kato, T. Adachi, H. Watanabe, R. Komaki, R. Kubo, T. Takada, and Y. Ikuma, "ATR CALL: A speech perception/production training system utilizing speech technology," in *Proc. ICA 2004*, 2004, vol. III, pp. 2319–2320.

[4] L. Hultzen, "Consonant Clusters in English," *American Speech*, vol. 49, no. 1, pp. 5–19, Feb. 1965.

[5] J. R. Anderson, *Rules of the Mind*. Hillsdale, NJ: Erlbaum, 1993.

[6] K. R. Koedinger, J. R. Anderson, W. H. Hadley, and M. A. Mark, "Intelligent tutoring goes to school in the big city," *International Journal of Artificial Intelligence in Education*, no. 8, pp. 30–43, 1997.

[7] M. Eskenazi, Y. Ke, J. Albornoz, and K. Probst, "The Fluency Pronunciation Trainer: Update and user issues," presented at the STiLL Workshop on Speech Technology in Language Learning, Marhollmen, 1998.

[8] M. Celce-Murcia, D. Brinton, and J. Goodwin, *Teaching Pronunciation*. Cambridge University Press, 1996.

[9] E. Altenberg, "The judgment, perception, and production of consonant clusters in a second language," *IRAL*, vol. 43, pp. 53–80, 2005.

Error Detection for Teaching Communicative Competence

W. Lewis Johnson
Alelo Inc.
Los Angeles, CA USA
ljohnson@alelo.com

***Abstract*— The primary goal of Alelo's language and culture products is to help learners develop communicative competence. This paper gives an overview of Alelo's instructional and technical approaches for developing communicative competence, and places pronunciation training within that broader context. Courses address a wide range of knowledge, skills, attitudes and other relevant factors pertaining to communicative competence, including pronunciation skills. Error detection and remediation play an important role; however they must be provided in a way that supports the broader goal of promoting communicative competence. Speech and language technology adapted for language learners provides a foundation for this work. Focused pronunciation activities, some of which require specialized speech models, support the learning process. Learner data is used to develop a profile of each learner's competencies and predict their future attrition and decay. This makes it possible to provide learners with individualized curricula focusing on their individual needs.**

Keywords-computer-aided language learning; speech technology; learner modeling; error detection and remediation

I. INTRODUCTION

The foremost objective of language learning is to communicate effectively in real-world settings and situations [2]. To achieve this, learners must acquire a variety of language-related knowledge, skills, abilities and other related factors (KSAOs). Pronouncing the sounds of the language is one of these skills, and one that is a common focus of computer-aided language learning. However it is not necessarily the most important skill. For example, the proficiency guidelines published by the American Council for the Teaching of Foreign Languages cite lack of accent as a criterion only at the highest levels of proficiency [1]. Below that level it is sufficient for learners to be understood by native speakers of the language. Various kinds of errors can impede understanding and lead to misinterpretation, including problems with grammatical structures and vocabulary, unusual phrasing, or failure to conform to the pragmatic norms of discourse in the language. Native speakers frequently misjudge misunderstandings of non-native speech as being due to poor pronunciation [7], which may exaggerate the perceived importance of pronunciation errors.

This paper gives an overview of the role of error detection and remediation within Alelo's instructional approach, which centers on the use of social simulations to help learners develop communicative skills. The approach is realized in a technology platform that is able to detect a range of language errors, both in the social simulations themselves and in other activities that help learners acquire the KSAOs that they will need to succeed in the social simulations. The paper discusses the role of learner language and learner errors in the underlying speech and language models. Then it focuses on methods for analyzing learner performance to provide feedback on pronunciation. Finally, the paper describes how learner performance data is used to assess communicative competencies in order to provide a learning experience customized to individual needs.

II. INSTRUCTIONAL APPROACH AND EXAMPLES

Alelo courses center on the use of social simulations for practicing and assessing communication skills. Social simulations engage learners in conversation with computer-generated characters. The computer characters interact with the learners in a manner appropriate to the social and cultural context of the conversation, and learners are encouraged to do the same.

This approach is incorporated into a variety of courses, in use around the world for language and culture education and training. Figure 1 shows an example of a dialog implementing the social-simulation approach, taken from a course in Tetum, the language spoken in East Timor. This course is intended for use by Australian military personnel preparing for overseas operations. In this dialog, the learner plays the role of a soldier named John Pearson (on the left) standing guard outside a restricted area. A computer-controlled character on the right, a Timorese man named Marco, wishes to enter the area. A conversation ensues. The learner engages with Marco by clicking on the Record button (top left) and speaking in Tetum. The system interprets the meaning of each learner utterance and the character responds accordingly.

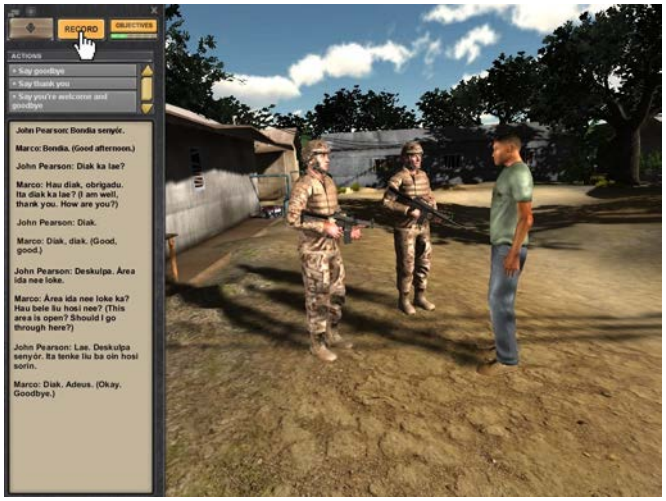


Figure 1. Example social simulation

In this example, the dialog began with basic greetings and rapport-building exchanges, as is customary in Timorese culture. The learner then attempted to inform Marco that the area is closed and off-limits. However the learner has made a mistake in using the word “loke” (open) instead of “taka” (closed), leading to confusion on Marco’s part. The learner is forced to correct himself and explain that the area is in fact closed.

The approach is motivated by research in how people learn in context [3]. The context affects how communicative skills are learned and how they are recalled and applied in real-world settings. The social-simulation approach therefore builds on the theory and methods of task-based language instruction [5]. As with other simulation-based learning approaches, the social-simulation approach is an experiential education method [4]. It gives learners opportunities to learn by doing and then seeing the results of what they did. Here, the learner had to explain to a man that he cannot pass, and saw the result of his mistake (the man asking for confirmation that he is free to enter). We refer to this type of error feedback as *organic feedback*, meaning that it is intrinsic to the behavior of the characters in the simulation. Learners find this to be a highly salient and meaningful way of receiving feedback on their performance.

At the conclusion of the simulation, the system generates a summary of the learner’s performance (Figure 2). This helps learners see the areas where they need to improve, and helps instructors to track learner progress so that they provide additional feedback and guidance. As this example illustrates, learners receive feedback on multiple aspects of their language performance. First, they get feedback on how well they performed the task, i.e., whether they accomplished the objective. They get feedback on the quality of their language performance, including how much they relied on hints, and whether they produced a variety of utterances instead of repeating the same memorized phrases over and over again. They also receive feedback on specific language errors. In this case, the focus is on improper use of vocabulary, a common problem for learners who are at the early stages of vocabulary mastery [11].

Figure 2. Example scenario feedback

Viewed in this context, pronunciation skill and pronunciation training play a supportive and distinctly secondary role in the simulation and in feedback. Learners do not get feedback on pronunciation from the characters in the simulation. It is unusual in real life for native speakers to critique learner pronunciation in the course of a conversation, and such feedback would tend to break the sense of immersion in the simulation and turn it into a pronunciation exercise. The structure of assessments such as Figure 2 is informed by the needs and preferences of language instructors. In the case of the Tetum course, Australian military instructors want to know firstly whether the learner is able to accomplish the task, and secondly their command of phrases, vocabulary, and grammatical forms in performing it. Pronunciation accuracy is relevant, but has lower priority than these other factors. This prioritization is consistent with common standards for world language instruction (e.g., [2]).

This is not meant to imply that pronunciation accuracy is insignificant. On the contrary, the social-simulation approach requires careful attention to the language spoken by learners, including common patterns of pronunciation errors, and techniques for preventing them and remediating them. Otherwise dialogs such as the one shown might easily break down because the computer is unable to understand the learner’s speech with sufficient robustness and reliability, and therefore cannot engage effectively in conversation with the learner.

III. MULTIMODAL COMMUNICATION WITH LEARNERS

The foundation of the Alelo technical approach is a computational architecture for multimodal communication, designed for use in learning applications. This architecture is employed in all activities that involve social simulation, including interactions with animated characters such as the one with Marco in Figure 1. The following is a brief overview of this model; further details may be found in [8].

Processing in the Alelo architecture is a continuous cycle of learner behavior interpretation, intent planning, and behavior generation. Behavior interpretation involves processing input from the learner and inferring the communicative intent, i.e., the meaning that the learner intended to convey. In the intent-planning phase, the system decides what action to take in response, typically a communicative action. Then, in the behavior-generation phase, the system determines how to perform the action. This architecture has much in common with other conversational agent architectures, such as the SAIBA architecture [15]. What makes it unique is that it is designed for use in teaching intercultural communication skills.

We take a broad, comprehensive view of intercultural communication, including both verbal and nonverbal skills. The architecture takes input from the learner through both a verbal and a nonverbal channel, and then interprets the combination in the context of the culture to arrive at a behavior interpretation. The medium used for each channel depends upon the capabilities of the computing platform, as well as the learning objectives of the particular activity.

Verbal input is commonly, although not exclusively, obtained through speech processing. We have designed the architecture so that it can still function if the sound input channel or the speech recognition module has been disabled on the learner’s computer. In such cases, learners may input their choices

from menus instead. We are also looking to support text input, to help learners develop written language skills, as well as improve their mastery of grammatical forms in the language.

The nonverbal channel is used to capture gestures and body movements that have a communicative role in the target culture. This includes hand-gesture greetings (e.g., the palm-on-heart gesture common in the Islamic world), bowing, shaking hands, etc. In current courses, learners select these from menus; however, input could also be performed through a motion-capture interface such as Microsoft's Kinect system.

The output of the behavior-interpretation phase is a communicative act that describes the intended communicative function of the learner's input, together with features of the input that are useful for analysis of learner performance, e.g., a transcription of the spoken utterance and its duration. Communicative functions play a central role in the system, since the primary learning objective is to develop communicative competence. The dialog system processes communicative acts in real time, allowing non-player characters to respond appropriately to each dialog move the learner makes in the conversation. It also records and logs them for analysis and learner modeling. The learner modeling system uses the evidence from the learner's behavior to assess the learner's mastery of each of the communicative competencies in the curriculum [9].

IV. Support for Learner Language

A key feature of Alelo's technical approach is that it is designed to process and understand learner language, i.e., language forms produced by learners [6]. All components of the architecture are designed with the characteristics of learner language in mind, particularly the language of novice-to-intermediate learners, who have been the most common users of Alelo products to date.

Learner language at this level tends to have relatively limited complexity, consisting of relatively short utterances. Learners at the novice level make frequent use of memorized phrases [1]. Learners tend to have a restricted vocabulary, which is specified in the course curriculum and therefore somewhat predictable. These factors, together with the task and dialog context, serve to constrain the complexity of the natural language the system needs to understand. Thus, for example, in the dialog context in Figure 1, Marco can expect the learner to engage in a relatively limited range of communicative functions, and express them in a limited number of ways.

At the same time, learner language can have a broad range of variability in terms of accent, pronunciation errors, and other errors in linguistic forms and usage. The behavior-interpretation system therefore must have sufficient tolerance for variability and sensitivity to error. Tolerance for variability needs to be sufficient to allow the system to successfully interpret the learner's speech in most cases and respond accordingly, particularly in a dialog context. Sensitivity to error is required in order to detect and classify learner errors, assess learner mastery of component linguistic skills, and provide constructive feedback.

The desired degree of tolerance and sensitivity depends upon the level of the course and the learning objectives of the particular learning activity. For beginners, the highest priority is building confidence and allowing them to experience success; therefore a high tolerance for pronunciation errors is important. As learners progress, the tolerance for errors should decrease, to encourage them to improve.

To achieve sufficient tolerance for variability in learner pronunciation, we train our speech recognition models using a mixture of native speech and learner speech. The incorporation of learner speech helps to ensure that the input system is relatively tolerant of variability in accent. The speech recognizer combines a language model built out of vocabulary and phrases from the course, and a "garbage model" that can match with low probability against any utterance. The garbage model ensures that each learner utterance is positively recognized with sufficient probability, thereby minimizing the occurrence of false recognitions.

The speech input system dynamically switches between language models as the learner progresses. As the learner advances to more complex material, the perplexity of the language model increases. This has the effect of progressively increasing the accuracy threshold for the learner's speech, since utterances need to be recognized with progressively higher probability to distinguish them from alternative phrases and from the garbage model.

Sensitivity to error is achieved by incorporating common learner errors into the language model. The choice of which errors to include depends on the objectives of the learning activity, the reliability with which errors can be detected, and what sort of feedback is appropriate in a given context. Since learning objectives cover a range of linguistic forms (vocabulary, phrases, and grammatical structures), functions (communicative functions and rhetorical structures), and practices (pragmatics and context-dependent determiners of usage), a variety of types of errors can occur, and these can potentially be captured in the language model. In practice we utilize such error models mainly in focused exercises involving specific communicative skills, and only to a limited extent in extended dialogs (as in Figure 1), since this would defeat the purpose of the latter. If we were to continually interrupt the dialog with feedback on grammar and pronunciation, for example, the activity would quickly cease to be an exercise in communication and become an exercise in grammar and pronunciation.

We have conducted evaluations of the performance of the spoken dialog system, and have reported the results elsewhere [13]. In [13] we evaluated the speech-understanding performance of our Subsaharan French course against human raters. The percentage of misunderstandings (where the system assigned an interpretation that was different from the expert raters' interpretation) was quite low, 3.5% of the conversational turns. When human raters judged utterances to be incomprehensible, the system also rejected them as "garbage" 95% of the time. However there were many utterances (33% of the total) that the raters found comprehensible, but the system rejected as garbage. In 63% of these cases, the learners had one or more pronunciation errors. After further analysis, we concluded that many of the learners were confused by French orthography, and so we corrected the problem by providing better spoken hints, not by changing the speech understanding algorithms.

At the same time, we want to ensure that learners and teachers have a positive subjective experience with the system. Do they feel that the speech input system has an appropriate degree of tolerance for error, so that the activity is neither too

easy nor too difficult? In general the answer is yes, with the exception of tone errors in tonal languages such as Chinese. Since our speech models are built from segmental phonemes, our speech recognizer can't distinguish tones in continuous speech, and is therefore overly tolerant of such errors. This requires us to make special provision for teaching the pronunciation of tones in tonal languages, as will be described below.

V. PART-TASK LEARNING

As noted above, successfully negotiating complex dialogs requires mastery of a combination of knowledge, skills, abilities, and other characteristics (KSAOs) pertaining to linguistic forms, functions, and practices. To help learners acquire these component KSAOs, Alelo courses provide a variety of structured part-task learning activities focused on a limited set of KSAOs. In these part-task exercises, learners receive much more detailed feedback on their performance of individual KSAOs, including pronunciation skills, than they do from extended dialogs.

Figure 3 shows a common type of part-task learning activity called a mini-dialog, which enables learners to practice individual communicative functions. This example is taken from the goEnglish course, an on-line course in colloquial American English developed for Voice of America. goEnglish is available in multiple languages and has over 100,000 registered users worldwide. Learning modules deal with a variety of situations in everyday life, work, and school in the United States.

Figure 3. A mini-dialog exercise

This example is taken from a module on ordering food in a fast-food restaurant. There are a number of communicative practices involved in this activity that may be unfamiliar to people from other cultures. For example, when one orders a hamburger, one may choose from a range of toppings and condiments. If the cook gets the order wrong, the customer will need to negotiate with the restaurant staff to get it corrected. Part-task learning activities in the module introduce some of the individual communicative skills that can be helpful in such situations.

In this example, the learner has ordered a hamburger without tomato, and the hamburger arrives with a tomato slice on it. The learner's task is to inform the counter worker of the error. The learner decides what to say and clicks on the record button to say it. The software evaluates the choice and gives a response. There is no single right way to complete the exercise. Any well-formed utterance that conveys the intended meaning and is appropriate to the situation is rated as acceptable.

In this example, the learner's input was "I asked no tomato." This utterance illustrates a common learner mistake, i.e., omitting a function word, in this case "for." The system detects the error and provides an explanation and feedback, presented in part by the Virtual Coach (top right), who pops up and comments on the learner's response. Feedback typically includes a cognitive component (evaluation of the learner's response) and an affective one (encouragement and mitigation of embarrassment). This approach builds on research showing that pedagogical agents that interact with learners in a socially appropriate way can promote positive learner attitudes and yield improved student learning outcomes [10].

To detect and respond to such errors, the speech processing system employs grammar-based language models that match a variety of appropriate and inappropriate responses. Errors in grammar, morphology, semantics, and pragmatics are captured in this fashion. Pronunciation errors can be captured here as well, although in practice we tend to cover pronunciation in other activities that focus specifically on those skills. And since, as described above, a garbage model is active to capture utterances that do not match any of the expected responses, learners also get feedback if the system cannot precisely pinpoint the error.

VI. PHONETICS AND PRONUNCIATION EXERCISES

To further support the learning process, we provide part-task learning activities that focus on particular linguistic forms, including the sounds of the language. Some activities give learners opportunities to practice listening to and discriminating different sounds. Others give them practice speaking the sounds. These activities help to reinforce the phonetic skills that learners are acquiring in the conversational exercises.

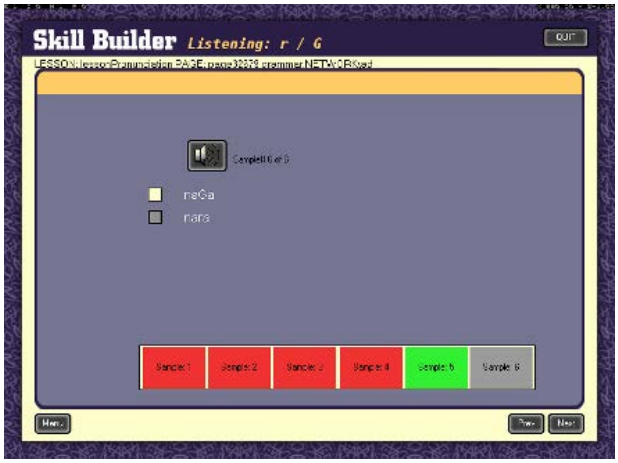


Figure 4. A sound discrimination exercise

Figure 4 shows an example listening exercise, taken from an Iraqi Arabic course. It gives learners the opportunity to practice distinguishing the apical rhotic /r/ from the voiced velar fricative /ɣ/, transliterated here as "G." English speakers often have difficulty distinguishing these sounds. Learners

listen to a series of words containing one or the other of these sounds and indicate which sound they hear. This practice helps make them aware of the differences in sounds and better able to discriminate between them.

Pronunciation practice activities give learners practice speaking the sounds of language. These also focus on the sounds that learners tend to confuse and have difficulty discriminating. Figure 5 shows one such pronunciation practice exercise for Iraqi Arabic, again focusing on /r/ and /ɣ/. Learners are presented with minimal pair words that differ only in the target sound. They hear a native speaker pronounce each word, then they attempt to pronounce them. The system rates how close each learner utterance is to the two alternatives, and provides graphical feedback on a moving slider (top center). As the learner repeats the exercise, the displays at the bottom show their cumulative performance in producing these sounds. They continue until they are able to produce the sounds with sufficient reliability.

These exercises employ acoustic models that are constructed specifically for discriminating such sounds. We collect recordings of both native Arabic speakers and language learners speaking the target minimal pair words and build the models from the recordings. This capability is still experimental, while we collect additional training data to increase recognition reliability.

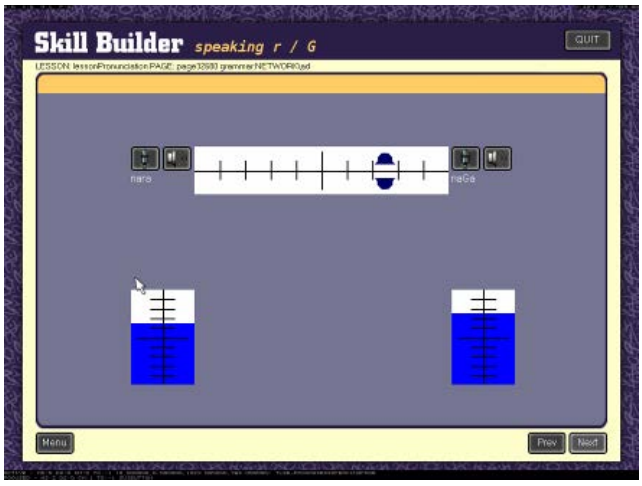


Figure 5. A phone practice exercise

One of Alelo's development partners, VIFIN (Videnscenter for Integration), has developed an additional pronunciation practice activity using Alelo technology and has integrated it into a course they developed using Alelo's SocialSim™ technology platform. This activity, called the Pronunciation Trainer, is shown in Figure 6. It is intended to help learners of Danish become familiar with its sounds. It presents the learner with a set of Danish words, each of which is an example of a particular phone in the Danish language. Learners repeatedly listen to and practice saying the words. A Danish speech recognizer developed collaboratively by VIFIN and Alelo attempts to recognize each word. A pedagogical agent named Harald (center right) provides feedback after each attempt. Each successful word recognition is treated as positive evidence that the learner has mastered the target phones, and each unsuccessful recognition is treated as negative evidence. The Pronunciation Trainer also serves as a reference tool. Learners can search not only words to practice pronunciation, but also single letters to find all words containing the letter with pronunciation variants.

Figure 6. VIFIN's Pronunciation Trainer

As mentioned above, tone production is a significant part of pronunciation in tonal languages such as Mandarin Chinese. Speech recognition algorithms typically process segmental phones and are insensitive to tones. So when we apply Alelo methods to tonal languages, we provide specialized pronunciation activities focusing on tone analysis and feedback.

Figure 7 shows the user interface for a prototype tone practice exercise called Tone Warrior. In this exercise, learners practice speaking two-syllable phrases, and are evaluated on their ability to produce accurate tones in these phrases. Two-syllable phrases expose learners to the complex interactions between the tones of adjoining syllables in languages such as Chinese, without introducing the added complexity of prosodic contours in continuous speech.

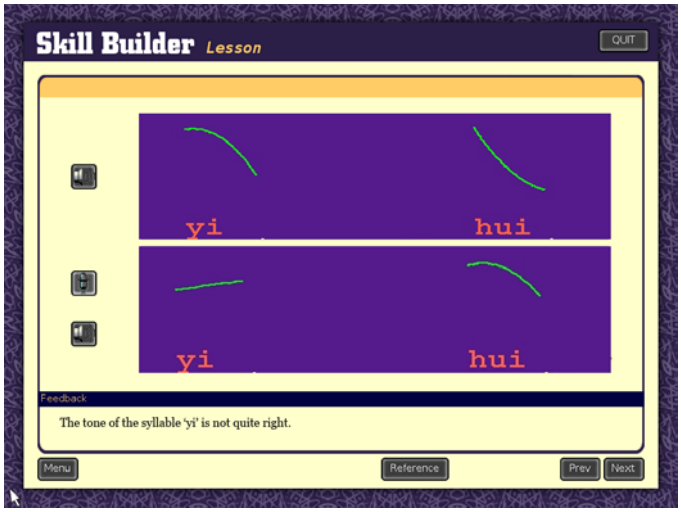


Figure 7. Tone Warrior pronunciation activity

Tones are represented by pitch or fundamental frequency (f_0) [14], and analyzed using a super-resolution pitch detection (SRPD) algorithm [12]. The interface presents smoothed pitch

contours that allow the learner to compare the shape of tones spoken by a native speaker with their own tones. The pitch detection algorithm can also distinguish qualitative tone shapes, such as the shapes of the four tones in Mandarin Chinese, and so can detect when the tone shape of a particular word is incorrect, as in this example.

VII. Conclusions

This paper has provided an overview of automated error detection and feedback in the context of Alelo's language and culture courses, and has placed pronunciation error detection and feedback in that context. Alelo's approach emphasizes communicative competence, consistent with commonly recognized proficiency standards. Alelo's speech technology is designed to support robust human-computer conversational interactions, in the context of social simulations. Pronunciation error detection plays a role in this context, alongside detection of other types of language errors, insofar as it supports the broader goal of promoting communicative competence.

Proficiency standards indicate that as learners advance, the accuracy of their pronunciation should also improve. Alelo's spoken dialog technology is consistent with this model, since it requires learners to produce more accurate language in advanced dialogs than in beginning-level dialogs.

As Alelo continues to develop its learning methods and supports more advanced levels of language proficiency, pronunciation error detection can play a more significant role. We therefore see an expanding role for pronunciation practice activities that complement conversational practice activities. There is also a role for pronunciation assessment within conversational practice activities, as a component of an overall summary of the learner's competencies. However we will continue to view pronunciation as just one skill among the many linguistic KSAOs that learners must master, in support of the overarching goal of promoting communicative competence.

Acknowledgment

The author wishes to thank the members of the Alelo team who helped with the preparation of materials for this article, including Jidong Tao, Rebecca Row, and Mickey Rosenberg. Projects described here were developed under sponsorship from the Australian Army Simulation Wing, Voice of America, USMC PM TRASYS, and other agencies.

References

[1] American Council on the Teaching of Foreign Languages, "ACTFL proficiency guidelines: Speaking, writing, listening, and reading," Alexandria, VA USA: ACTFL, 2012.

[2] American Council on the Teaching of Foreign Languages, "Standards for foreign language learning: Preparing for the 21st century," Alexandria, VA USA: ACTFL, 2012.

[3] J.D. Bransford, A.L. Brown, and R.R. Cocking, "How people learn: Brain, mind, experience, and school," Washington, DC: The National Academies Press, 2000.

[4] J. Dewey, "Experience and education," New York: Collier Books, 1938.

[5] R. Ellis, "Task-based language learning and teaching," Oxford: Oxford University Press, 2003.

[6] R. Ellis and G. Barkhuizen, "Analyzing learner language," Oxford: Oxford University Press, 2005.

[7] S.M. Gass and L. Selinker, "Second Language Acquisition," New York: Routledge, 2008.

[8] W.L. Johnson, L. Friedland, A.M. Watson, and E.A. Surface, "The art and science of developing intercultural competence," in P.J. Durlach and A.M. Lesgold (Eds.), "Adaptive technologies for training and education," pp. 261-285, New York: Cambridge University Press, 2012.

[9] W.L. Johnson and A. Sagae, "Personalized refresher training based on a model of competency acquisition and decay," in Proceedings of the 2nd International Conference on Applied Digital Human Modeling, in press.

[10] W. L. Johnson, and N. Wang, "Politeness in interactive educational software," in C. Hayes & C. Miller (Eds.), Human-Computer Etiquette, London: Taylor & Francis, 2010.

[11] B. Laufer-Dvorkin, "Similar lexical forms in interlanguage," Tübingen: Narr, 1991.

[12] Y. Medan, E. Yair, and D. Chazan, "Super resolution pitch determination of speech singals," *IEEE Trans. ASSP*, vol 39, pp. 40-48, 1991.

[13] A. Sagae, W.L. Johnson, and S. Bodnar, "Validation of a dialog system for language learners," in Proceedings of the SIGDIAL 2010 Conference, pp. 241-244. Tokyo: Association for Computational Linguistics, 2010.

[14] Ye Tian, Jian-Lai Zhou, Min Chu, and Eric Chang, "Tone recognition with fractionized models and outlined features," in Proceedings of *ICASSP* 2004, Quebec, Canada, pp. 105-108, 2004.

[15] H. Vilhjalmsson and S. Marsella, "Social Performance Framework," in Proceedings of the AAAI Workshop on Modular Construction of Human-Like Intelligence, Menlo Park: AAAI, 2005.

Why and How Our Automated Reading Tutor Listens

Jack Mostow
Project LISTEN (www.cs.cmu.edu/~listen), School of Computer Science
Carnegie Mellon University
Pittsburgh, PA, USA
mostow@cs.cmu.edu

Abstract— **Project LISTEN's Reading Tutor listens to children read aloud, and helps them learn to read. This paper outlines how it gives feedback, how it uses ASR, and how we measure its accuracy. It describes how we model various aspects of oral reading, some ideas we tried, and lessons we have learned about acoustic models, lexical models, confidence scores, language models, alignment methods, and prosodic models.**

Keywords: speech recognition; reading tutor; oral reading

I. INTRODUCTION

Automated reading tutors [1-4] use automatic speech recognition (ASR) to listen to students read aloud. American children typically read aloud in grades 1-2 (ages 6-7) and are expected to be fluent silent readers by grade 4, often called the transition from "learning to read" to "reading to learn."

Reading is more than turning text into speech; its goal is to making meaning from print. Thus reading requires the ability to map graphemes to phonemes; decode new words; identify familiar words quickly; read connected text quickly, accurately, effortlessly, and expressively; retrieve context-appropriate word senses; comprehend the meaning of text; and stay motivated enough to practice reading and build fluency.

From the viewpoint of speech recognition, children's oral reading is often marked by hesitations, false starts, miscues (reading mistakes), regressions (rereading one or more words), list-like prosody, and off-task speech. Deviations from a dictionary pronunciation of a text word include mistakes in decoding, identifying, or pronouncing the word, dialect phenomena, and individual speech defects, such as the inability to produce or distinguish certain phonemes. As readers gain fluency, they hesitate less often, regress less, make fewer miscues, and read faster and more expressively.

A note about terminology: to reduce the potential for confusion, the word "mistake" refers in this paper to incorrect reading or pronunciation by the child; the word "error" refers to incorrect listening by the computer.

This paper is organized as follows. Section II describes why the Reading Tutor listens. Section III defines our measures of how well it listens. Section IV discusses how we represent and train the models it uses to listen. Along the way, we discuss some of the approaches we tried over the past 20+ years, and lessons we learned. Finally, Section V concludes.

II. PURPOSES OF LISTENING IN A READING TUTOR

The Reading Tutor listens for several purposes. By detecting speech and silence and using timing information, it decides when and how to respond. By aligning the ASR output with the text, it tracks the reader's position in the text. By comparing each text word with the hypothesized word aligned against it, it detects oral reading miscues. By analyzing the time alignment of the ASR output, it computes how long the student takes to identify each word and read it aloud. By extracting the pitch, amplitude, and duration of read words, it computes their prosodic contour. We mine these various sorts of information off-line to assess students and evaluate tutor actions, but this paper is about the speech information the Reading Tutor uses at runtime.

A session with the Reading Tutor starts when the child clicks *Hello* and uses a talking menu interface to log in by clicking on his or her name and (as a light-weight but easy-to-remember password) birth month. The Reading Tutor then takes turns with the child at picking a text to read or other activity to do, such as jointly composing a story. The session ends when the child logs out by clicking an on-screen *Stop* sign, or times out by not speaking or clicking for 30 seconds, or if the Reading Tutor crashes or hangs.

Reading Tutor activities are built out of several types of steps, each with its own screen interface: assisted oral reading (and narrating); tutor instruction; multiple choice questions; keyboard input; using on-screen letter tiles to build words and sound them out; and free spoken responses.

Project LISTEN has focused primarily on assisted oral reading, and so does this paper. Some other types of steps also involve listening. In word-building steps, the Reading Tutor prompts the child to sound out the word, and tries to

The research reported here was supported by the Institute of Education Sciences, U.S. Department of Education, through Grants R305B070458 and R305A080628, and by the National Science Foundation under ITR/IERI Grant No. REC-0326153. Any opinions, findings, and conclusions or recommendations expressed in this publication are those of the author and do not necessarily reflect the views or official policies, either expressed or implied of the Institute, the U.S. Department of Education, the National Science Foundation, or the United States Government. I thank the educators and students who helped generate our data, and the many LISTENers over the years who co-authored the work summarized in this paper and cited in the References.

follow along, but does not attempt to detect mistakes. In free-response steps, the Reading Tutor graphically indicates the approximate amount of speech, but records it without trying to recognize it at runtime. However, we've worked on recognizing some types of speech off-line [5-7].

The Reading Tutor reacts to speech, mouse clicks, and delays by responding with graphical and spoken feedback, described respectively in Sections II.A and II.B.

A. *Graphical interface*

Assisted oral reading uses a graphical interface. As the screenshot in Figure 1 shows, a robot persona provides a visible audience by blinking sporadically to appear animate, gazing at the current word to appear attentive, and displaying a volume meter to show that it's listening. Up and down buttons adjust its output volume. (We hide the input level control to protect it from misadjustment.) Navigation buttons at the top of the screen consist of *Stop* (to quit the story) and *Go* (to advance to the next sentence). The Reading Tutor displays text on a book-like background by adding one sentence at a time and graying out the previous sentences, unlike educational software that displays text page by page like a book.

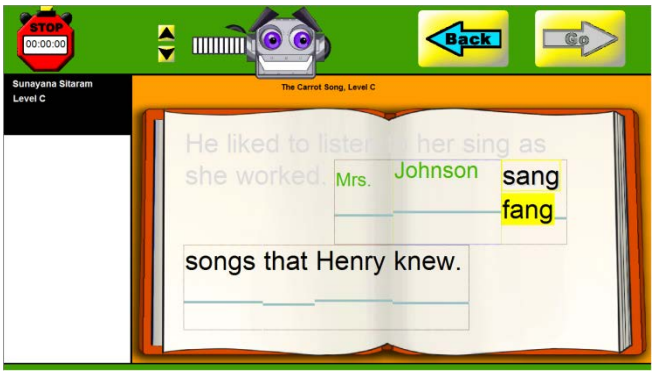


Figure 1: Reading Tutor screenshot (2012)

The Reading Tutor displays text sentence by sentence for three reasons. One reason is ASR accuracy. Controlling which sentence is displayed imposes a strong constraint on what the student can read aloud. A second reason is usability of the spoken dialogue. Before displaying the next sentence, the tutor has an opportunity to intervene without risk of interrupting the student. A third reason is pedagogical. Postponing the display of the next sentence frees the tutor to decide on the fly what to display next, e.g. to help decode a hard word, explain unfamiliar vocabulary, or give comprehension assistance.

The Reading Tutor maps various components of its internal state to graphical properties, such as word color, background color, shadowing, and underlining. It displays earlier sentences in gray, words to read in black, credited words as green, and future sentences in white, i.e. invisible. It shadows the word it thinks the child will read next, underlines a word to prompt the child to read it, boxes the word the cursor is on, and highlights the background of a word in yellow while reading it aloud or hinting how to do so, which may involve temporarily showing a rhyming or other related word below it. Figure 1 shows the Reading Tutor saying "rhymes with *fang*" as a hint to decode *sang*.

The Reading Tutor can display assisted reading rate on a "readometer" while the child is reading a story, and in an on-screen certificate after the story as a reward for finishing it. In addition, it can provide real-time graphical feedback on the child's oral reading prosody [8], for example by mapping the loudness of a word read by the student to its size, its pitch to its vertical position, and the narrator's pitch contour to a staircase-like sequence of lines. Thus in Figure 1, *Mrs.* is smaller because it was spoken softly, *Johnson* is higher because it was spoken with rising inflection characteristic of a question or guess, both words are green because the Reading Tutor has credited them as read, and the color, size, and position of the remaining words are unchanged because the child hasn't read them yet.

B. *Multimodal dialogue*

The Reading Tutor's dialogue architecture [9, 10] is driven by speech, silence, time, and mouse clicks. It represents turn-taking state in terms of four binary variables:

- Is the student speaking?
- Is the Reading Tutor speaking?
- Does the student have the floor?
- Does the Reading Tutor have the floor?

Each variable has a timer that records when it last changed.

At any point in time the student, the Reading Tutor, both, or neither may have the floor. Transitions between states occur when the student or Reading Tutor starts or stops speaking, or when one of the timers reaches a specified threshold value. The states and timers govern whether, when, and how the Reading Tutor speaks.

For example, after a 2-second silence, the Reading Tutor may backchannel to encourage the student to continue reading. However, if the student remains silent for 2 additional seconds, the Reading Tutor takes the floor to verbally prompt the student to click for help.

Backchanneling does not take the floor away from the student. For instance, if the Reading Tutor detects a skipped word, it underlines the word and coughs to draw attention to it, but it's still the student's turn.

Usually the Reading Tutor does not take the floor when the student has it. An exception is choral reading, when the Reading Tutor prompts the student to "read with me."

The Reading Tutor takes the floor when the child clicks the mouse, whether on *Stop* (to quit the story), *Go* (to advance to the next sentence), a word (to get help reading it), below the sentence (to hear the Reading Tutor read it), or elsewhere on the screen (by mistake).

By design, the Reading Tutor responds to all clicks rather than ignore the child, even if only to explain why it can't perform the requested action. For instance, if the child clicks on the *Go* button without reading at least half the sentence, the Reading Tutor says "Sorry, can't go on right now." To make its behavior easier to understand, the Reading Tutor responds to mouse clicks immediately to

make clear what it is responding to. Thus it interrupts itself if it is speaking, rather than wait to finish what it is saying. Waiting would complicate how it represents and displays the dialogue state.

The Reading Tutor also takes control when it hears the student read the end of the sentence. If any content words remain uncredited, it waits for the student to read. If it heard the student read the entire sentence fluently, it advances to the next sentence without intervening. If not, it reads the sentence aloud first, so as to scaffold comprehension, because failure to read the sentence fluently indicates that the child may not have understood it.

The Reading Tutor uses the overall distribution of inter-word latencies [11] to estimate the child's reading level [12], which it uses in deciding which stories to pick from.

In short, listening to a child read aloud enables the Reading Tutor to decide when and how to give feedback, track the child's position in the text, compute the latency before a word and the time to read it, detect miscues, assess children's oral reading fluency, and mirror their oral reading prosody. We next discuss ways to define and measure the accuracy of its listening for these purposes.

III. Listening Accuracy Metrics for Oral Reading

Over the years, we have evaluated the Reading Tutor's listening accuracy in several different ways.

A. Miscue detection accuracy

At first we focused on measuring accuracy in detecting oral reading miscues. Conventional word error rate can measure accuracy in *recognizing* miscues – i.e. in transcribing them. However, word error rate does not measure accuracy in *detecting* miscues [13].

The Reading Tutor detects miscues by aligning the hypothesis output by the ASR against the sentence displayed. Thus detecting a miscue does not require the ASR to recognize it correctly, merely to recognize it as anything other than the word the child was supposed to read.

We have measured miscue detection at different levels. At the highest level, which we call "text space," we treat miscue detection as a classification problem: classify each text word as read correctly, misread, or omitted, or simply as read correctly or not [14]. At this level, we define a miscue as a text word the child failed to read in the course of reading the sentence. This criterion treats false starts, sounding out, incorrect attempts, and other insertions as steps toward the goal of reading the text word, not as mistakes to remediate if they culminate in reading it correctly. This pedagogical policy means that ASR insertion errors don't matter except if hallucinating a word causes the Reading Tutor to misclassify it as read correctly, or leads the ASR astray, causing it to make deletion or substitution errors.

At the next level, which we call "speech space," we classify each *transcribed* word instead. Thus at this level each failed attempt to read a word counts as a miscue whether or not the child subsequently read the word correctly. These misreadings provide a welcome source of training and test data, since text-space miscues are scarce.

At the most detailed level, which we call the "time domain," we classify *time-aligned* transcript words. Time domain accuracy is more stringent. For instance, scoring an accepted word as true requires that it occur in approximately the same time interval in the time-aligned ASR output as in the time-aligned transcript.

B. Tracking accuracy

More recently we have measured tracking accuracy as well. These measures evaluate the Reading Tutor's estimate of the child's position in the current sentence. By aligning ASR output or a reference transcript of the child's oral reading against the sentence, we obtain a *trace*: a sequence of integer positions in the sentence, where position *i* represents the i^{th} word of the sentence. The sign of the integer encodes whether the aligned word matches the text word: + if yes, – if no.

The minimal edit distance between traces based on the reference transcript and the ASR output is a "speech space" measure of tracking error, defined as the number of insertions in, substitutions for, and deletions from the trace based on the reference transcript to turn it into the trace based on the ASR hypothesis.

For example, consider these alignments of transcribed and recognized readings to the text "Once upon a time, the dog":

```
Ref.: once up upon the time      dog
      +    –  +    –   +         +
Text: Once1 upon2   a3  time4,the5 dog6
      +     –      +   +    +    +
Hyp.: ONCE /AH_P/  A   TIME THE  DOG
```

Here "+" and "–" show if the aligned word matches the text. The transcript-based trace is +1, –2, +2, –3, +4, +6, where –2 comes from misreading "upon" as "up," and –3 comes from misreading "a" as "the." The hypothesis-based trace is +1, –2, +3, +4, +5, +6, where –2 comes from recognizing "up" as a truncation of "upon."

Aligning the two traces to minimize edit distance yields the sequence +1/+1, –2/–2, +2/, –3/+3, +4/+4, /+5, +6/+6 where:

+I/+I = accepted reading
–I/–I = detected miscue at correctly tracked position
+I/–I = false alarm at correctly tracked position
–I/+I = undetected miscue at correctly tracked position
/+I = inserted text word
/–I = inserted miscue or garbage
+I/ = deleted text word
–I/ = deleted miscue or garbage
+I/+J = mistracked accepted word
–I/–J = mistracked miscue
+I/–J = mistracked false alarm
–I/+J = mistracked undetected miscue

Thus +1/+1, –2/–2, +2/, –3/+3, +4/+4, /+5, +6/+6 shows that the transcript and hypothesis agree that the reader read $Once_1$ and misread $upon_2$. Then the ASR omits the correct

rereading of $upon_2$, and accepts the word a_3 rejected by the transcript. They agree that $time_4$ and dog_6 were read correctly, but the ASR hallucinates the word a_5.

"Time domain" measures of tracking accuracy take into account whether transcribed and hypothesized words occur at the same time in the speech signal [15]. Such measures compare time-aligned traces to determine the relationship between the transcript and hypothesis [16]. Each segment of a time-aligned trace is either a silence (#), a word that is aligned against a matching text word (+), or a word that is not (–).

A transcript segment and hypothesis segment that overlap in time have one of three temporal relations (labeled as shown). The midpoint of each one can fall within the other (=). The midpoint of the transcript segment can fall within the hypothesis segment, but not vice versa (<). Otherwise, the transcript segment contains the midpoint of the hypothesis segment, but not vice versa (>).

Finally, if transcribed and hypothesized segments that overlap in time are not silences, they may or may not be aligned to the same text word. We mark the latter case "J", short for the I/J notation used above for speech space.

To illustrate this notation, here it is for the fragment above:

```
1 Once (+=+)  ONCE
2      (#=#)
3 up   (-=-)  /AH_P/
4      (#<#)
5 upon (+<#)
6 the  (-=+)  A
7      (#=+)  TIME
8 time (+=#)
9 dog  (+=+)j THE
10     (#=+)  DOG
```

Transcript segments 1-3 match the times and text positions of the first three hypothesis segments. Segment 5 has an ASR deletion error: where the transcript contains "upon", the hypothesis contains a continuation of the preceding silence. In segment 6, the transcribed and hypothesized words have matching times and text positions, but the hypothesized word matches the text while the transcribed word does not. In segments 7 and 8, the transcript and hypothesis have the same word, but at different times. In segment 9, the transcribed and hypothesized words match different text words. Segment 10 has an ASR insertion error: where the transcript has silence, the hypothesis has the word DOG.

One time domain measure of tracking accuracy is how often (as a percentage of time) the position computed by the Reading Tutor based on the ASR output agrees with the child's position at the same point in time according to the transcript. Another measure is the average absolute distance (in words) between the two positions.

Off-line measures of tracking accuracy [16] are based on the final hypothesis output by the ASR at the end of the utterance. In contrast, real-time measures of tracking accuracy are based on the partial hypotheses output by the ASR as the child reads, and can therefore be considerably lower than off-line measures. The accuracy of real-time tracking trades off against its timeliness. Waiting as little as 0.2 seconds to estimate the reader's position yields a substantial increase in its accuracy [17].

To understand accuracy better, we wanted to distinguish regions of oral reading from regions of off-task speech. To identify off-task speech automatically in a transcript, we defined *deviation length* as the number of consecutive transcribed words without two successive matches to the text words aligned against them. By inspecting deviations of different lengths, we determined that deviations longer than 2 were nearly always off-task speech. Not surprisingly, tracking accuracy is much higher during on-task than off-task speech.

C. Accuracy of confidence metrics

A confidence metric estimates the probability or other score of whether an ASR word hypothesis is correct, or of whether the child read a word correctly. We measure the accuracy of the confidence metric by binning it, say into percentiles, and correlating the percentile for each bin against the actual percentage of words in each bin correct according to the reference transcript.

D. Indirect measures of accuracy

Besides the direct measures of listening accuracy discussed above, we have tested the Reading Tutor's listening indirectly by its ability to predict other measures, such as children's help requests [18], performance on cloze questions [19], and scores on paper tests of oral reading fluency [12], word identification [20], and comprehension [21, 22]. We measure predictive accuracy as correlation of predicted to actual scores, reaching 0.9 in some cases.

E. Micro-efficacy

A fine-grained test of the Reading Tutor is the impact of its instruction and practice on children's fluency in reading the taught or practiced word. We have used inter-word latency [11, 23] and word reading time as micro-measures of oral reading fluency at the level of individual words. We have used two types of methodology to test micro-efficacy.

An "invisible experiments" methodology inserts within-subject randomized controlled trials of alternative tutor actions in the Reading Tutor, with latency or reading time as the outcome of each trial, and aggregates the outcomes of such trials over many students and words [24]. One such experiment used thousands of trials to compare the impact of different ways to preview new words before a story on accuracy in reading them after the story [25]. Another experiment used over 180,000 trials to compare different forms of help on words based on how often the child read the word fluently at the next encounter [26, 27].

A model-fitting methodology uses data logged by the Reading Tutor to compare different types of practice. We have mostly used two classes of model.

A Dynamic Bayes Net model of knowledge tracing [28] estimates the value of a practice type as the probability of the student learning a word from an encounter of that type.

For instance, Beck [29] used this method to tease apart the immediate scaffolding effects of tutor assistance on performance from its subsequent effects, finding a small but statistically significant contribution of help to student learning.

The learning decomposition method [30, 31] uses an exponential decay model of word reading time to estimate the relative value of each type of practice as a coefficient on the number of encounters of that type. For instance, a learning decomposition comparison of rereading vs. new reading found that seeing a word again in a sentence seen before was worth only about half as much as seeing it in a new sentence. Other learning decomposition analyses compared massed vs. distributed encounters of a word, reading text chosen by the child vs. by the reading tutor [32], and transfer to similar words [33, 34]. More recently, we have used linear mixed effects models to predict the log of reading time so as to account properly for statistical dependencies on students, words, and stories by modeling them as random effects.

F. Macro-efficacy

The ultimate test of the Reading Tutor is its impact on the reading proficiency of the children who use it. To measure this impact, we have performed controlled studies to compare children's pre- to post-test gains on tests of various reading skills from using the Reading Tutor compared to gains from other treatments, including classroom instruction, other software, independent reading practice, and individual human tutoring [35-37]. So have other researchers [38-42]. We measure the Reading Tutor's impact on a tested skill as an effect size: the difference between the mean test score gains for two treatments, divided by the within-treatment standard deviation. Effect scores of 0.3 are considered small, 0.5 medium, and 0.8 large [43]. The Reading Tutor's effect sizes for fluency gains reached as high as 1.3 standard deviations in some studies, varying by comparison condition and student population, with English language learners apparently benefitting the most.

It is natural to ask how lower-level listening accuracy in tracking the reader and detecting miscues affects the Reading Tutor's educational efficacy. Unfortunately, this question is more readily asked than answered. Comparing the macro-efficacy of Reading Tutor versions that differ in the accuracy of their listening would be costly in time, money, and sample size. Analyzing the effects of deliberate listening errors on micro-efficacy might be more feasible, but it is far from clear that such effects are local. For instance, frustration caused by listening errors might be cumulative. How much low-level listening accuracy affects educational efficacy remains a question for future research.

IV. Automated Analysis of Oral Reading

The Reading Tutor uses Sphinx2 [44] to recognize read words, signal processing to extract their pitch and amplitude, and post-processing to support feedback and assessment.

The Reading Tutor's key ASR components include its acoustic-phonetic models, pronunciation lexicon, acoustic confidence scores, language models, and alignment methods. We now describe how we have represented and trained each of these models over the years, sometimes in a series of different ways.

A. Acoustic models

Project LISTEN originally used semi-continuous HMM acoustic models trained on adult female speech [14, 45]. After recording a small corpus of children's oral reading in a Wizard of Oz experiment, we used it to adapt the codebook means of our models [46]. Once we had a larger transcribed corpus of children's oral reading in the Reading Tutor, we trained HMMs on it from scratch. We trained continuous models once computers became fast enough to use them to recognize oral reading in real-time.

We had much less oral reading manually transcribed than not, so we tried training on untranscribed speech. We knew the text sentence that each utterance was an attempt to read, and we used cherry-picking heuristics to select the utterances likely to be or contain correct readings [47]. The resulting models performed better on a test set of children's oral reading recorded under similar conditions than training on the manually transcribed KIDS corpus [48, 49] of comparable size (approximately 5,000 utterances), collected under more controlled conditions in a quieter environment.

Despite this promising result, once we had accumulated a manually transcribed corpus of tens of thousands of oral reading utterances recorded by the Reading Tutor during normal use, automatically labeled data did not help; in fact, it actually hurt ASR accuracy when used to augment the manually transcribed training data. That is, ***quality trumps quantity***.

B. Lexical models

The Reading Tutor's active lexicon changes from sentence to sentence, taking advantage of knowing which sentence is currently displayed. The lexicon contains the words in the sentence. Their pronunciations come from CMUDICT [50] if it contains them, otherwise from the pronunciation component of a speech synthesizer.

The lexicon also contains distracters to model misreading and false starts. Over the years we have experimented with several types of distracters.

The only distracters we still use are the first kind we tried, namely phonetic truncations of the sentence words. For a word *w* whose pronunciation is *n* phonemes long, we add a distracter START_*w* with multiple pronunciations. They consist of initial subsequences of the *n* phonemes, containing at least the first 2 phonemes and at most *n*-2. Adding the truncation distracters increased miscue detection without increasing the false alarm rate (correctly read words misclassified as miscues). The resulting ASR detected about half the miscues rated by a human judge as serious enough to threaten comprehension, which in turn constituted only about half of the words whose transcription differed from the text – the more stringent criterion we used in our

later evaluations. The ASR rejected about 4% of correctly read words [46]. We prioritize accepting correct reading over detecting miscues, because children read 90% of words correctly unless the text exceeds their frustration level [51]. Also, rejecting a correctly read word frustrates the child, whereas accepting a miscue at worst confuses the child, though it may reinforce mislearning.

We initially included pronunciations with the first *n*-1 phonemes as well, but they reduced ASR accuracy by getting recognized too often in place of a correctly read word. One reason was a dialect phenomenon common among the children in our sample, namely dropping final consonants, such as /S/ at the end of a plural noun like *cats* or present tense verb like *sits*. Such a truncation may be a pronunciation mistake, but it does not constitute an oral reading miscue if it's the reader's normal pronunciation of the word. We therefore tried adding such truncations as alternate pronunciations for the correct word, but they reduced ASR accuracy by making it too easy to hallucinate. Accordingly, we do not include the first *n*-1 phonemes as a pronunciation, either of the correct word or as a distracter. This change remains our only accommodation to dialect phenomena.

The ASR often accepted misread short sentences as read correctly. The reason is that the ASR maps oral reading to the sequence of sentence words and distracters that it most resembles. Consequently, it typically does not detect a miscue unless the miscue resembles either a distracter for the correct word, or another word in the sentence, more than it resembles the correct word. A short sentence has fewer words for a miscue to resemble.

In an attempt to compensate for this limitation, we needed some additional distracters to help model miscues. We didn't want them to be too easy to hallucinate, so we refrained from adding individual phones as distracters. Instead, for short sentences we added as distracters a few two-syllable words used to spell out words over noisy radio connections: *alpha*, *bravo*, etc. Unfortunately, although they helped detect more miscues, they also hallucinated more miscues, so we wound up taking them back out.

Next we took a more systematic approach to miscue detection. By predicting likely miscues, we hoped to increase miscue detection without increasing false alarms. We explored three methods for predicting likely miscues.

The first method worked at the level of individual letter sounds, or more precisely graphophonemic mappings. For years, renowned reading researcher Richard Olson and his University of Colorado colleagues had been comparing the reading difficulties of identical and fraternal twins in order to quantify their genetic component. In the process they had recorded, phonetically transcribed, and annotated hundreds of twins' oral readings, in the process accumulating a database of tens of thousands of oral reading miscues. As a group project in a graduate course in machine learning, Fogarty *et al.* [52] mined this corpus to discover "malrules" that predict decoding mistakes at the level of individual graphophonemic mappings. Each malrule predicted that a grapheme *G* that should be decoded as some phoneme *P* would instead be decoded as some other phoneme *P'*. The 10 most frequent malrules turned out to be insertions and deletions. For instance, the two most frequent $G \rightarrow P \rightarrow P'$ rules were *s* → /S/ → _ and *s* → /Z/ → _, where _ denotes the empty string. These malrules predict deletion of the plural endings of *plant*s and *arm*s, respectively.

The other two methods [53] exploited the fact that most miscues consist of misreading one word as another. The "rote" method simply identified misreadings made by two or more readers on the 100 most frequent words in the corpus, and predicted that those misreadings would continue to occur. The "extrapolative" method generalized the relation between words and real-word misreadings of them, and predicted analogous misreadings of other target words.

Unfortunately, adding distracters other than the truncations targeted just the specific predicted miscues. They might detect a few more miscues with slightly fewer false alarms, but they increased the miscue detection rate significantly only by also increasing the false alarm rate [54]. In short, ***distracters detract.*** We therefore gave up on distracters to look for a more generic way to detect miscues.

C. Confidence scores

To detect miscues without specifically predicting them in advance, we tried a confidence metric approach. One metric [55] trained decision trees using three types of features.

Decoder-based features used word-level information from the ASR output, namely "log energy normalized by number of frames, acoustic score normalized by number of frames, language model score, lattice density, averaged phone perplexity, and duration."

Alignment-based features used contextual information about the target text word from the alignment of the ASR output against the sentence, such as whether the ASR accepted the word, the latency preceding the word, the number of previous or subsequent text words hypothesized in a row, and the average distance between hypothesis words aligned against the target word.

History-based features used information logged by the Reading Tutor about the student. Word-level features included how many times the student had encountered the target word in the past, how many of them were accepted, and the student's average latency before words in general. Utterance-level features of the current sentence included the number of utterances so far, and averaged over them, the number of words attempted, the number accepted, the number of jumps, and the number of regressions to the start of the sentence.

This method trained two decision trees that operated in "text space." The first decision tree estimated the probability that an accepted word was actually misread, based primarily on (i.e. using in the top two levels of the decision tree) phone perplexity, log energy, and acceptance by the ASR. To undo ASR deletion errors, the second decision tree estimated the probability that a rejected word was actually read correctly, based primarily on the number of successive text words preceding and following it.

With a training set of 3714 utterances and a test set of 1883 utterances by different children, and a baseline of 56% miscue detection and 4% false alarm rate, the method could either increase miscue detection to 59% or reduce the false alarm rate to 3%. These miscue detection rates are inflated due to treating unattempted words as deletions, so their actual values aren't meaningful, but the changes to them still show improvement.

By 2007, we had reduced the false alarm rate below 1%, with 23% detection of substitution miscues defined as a mismatch between the spoken word and the text according to the manual transcript. Since "text space" miscues are much rarer than correctly read words, we decided to evaluate "speech space" accuracy so as to measure ASR performance more sensitively. Tracking error, defined as the combined substitution and deletion rate in speech space, was below 2%, but the insertion rate was almost 17%.

We tried using a more conventional (i.e. speech space) acoustic confidence metric [56, 57] to filter ASR output. The confidence threshold ROC curve for the tradeoff between false positives and true positives exceeded 0.83 AUC (Area Under Curve). We expected a confidence metric to be a good way to decide whether a recognized word was in fact read correctly, because in principle, it should be able to detect miscues without relying on the language model and lexicon to predict them in advance. However, in practice, using a confidence metric to reject misread words is limited by tracking accuracy, because when the ASR goes off-track and recognizes a different word than the one the reader was trying to read, its confidence score is irrelevant – akin to closing the barn door after the cows have escaped. This inconvenient truth defeated our grand scheme to estimate the probability that a word was read correctly by combining acoustic confidence with other information such as a model of the student. That is, ***tracking trips up scoring*.**

D. Language models

The Reading Tutor uses a simple probabilistic finite state model of oral reading, which it generates on the fly for each sentence before displaying it [46]. In state *i*, it expects word *i* of the *n*-word sentence (with PrCorrect), a truncation of word *i* (with PrTruncate), a premature end of the utterance (with PrEndEarly), or a jump to state *j*, with different probabilities depending on *i* and *j*. A file specifies probabilities for the parameters PrCorrect, PrTruncate, PrEndEarly, PrRepeat, PrSkip, PrRestart, PrJumpBack, PrJumpForward, etc.

Initially this model was approximated as a bigram model. ASR accuracy improved when Ravi Mosur extended Sphinx2 to input finite state models and use them top-down. In contrast to bottom-up recognition, which relied on a lexicon-driven recognizer to hypothesize words, the top-down recognizer enabled high language model probabilities to overcome poor acoustic scores of words that the bottom-up method would have failed to recognize in the first place.

A classifier learning approach [58] reduced the speech space tracking error by adjusting language model probabilities iteratively. At each iteration, it used the language model from the previous iteration to recognize a training set of oral reading utterances, aligned the ASR output for each utterance against the target sentence to compute a trace, and scored it against the trace based on the transcript. It applied a simple credit assignment heuristic [59] to transitions between successive words in the recognized trace, classifying transitions that stayed on track as positive, and transitions that led off-track as negative. After using LogitBoost to learn a classifier from the labeled transitions, it increased the probability on transitions classified as positive, decreased the probability on transitions classified as negative, and used the adjusted language models to re-recognize the utterances. It repeated this cycle until tracking error started to rise. This method reduced tracking error from 9% to 7%, but was impractical to incorporate in the Reading Tutor because it involved applying the entire sequence of learned classifiers to the initial language model.

We explored various alternatives to simple *n*-state models. A key question was which additional states to include. For instance, the "watermark" model used $O(n^2)$ states of the form (i, j) to represent the reader being at word i and having previously read as far as word j. After attempts to design better finite-state models by hand, we extended Sphinx2 to allow non-finite-state models, and let a user-defined function directly compute the probability $\Pr(w \mid h)$ of word w following the preceding sequence h of recognized words. A SVM trained on such features as the frequencies of different transition types in h yielded a language model that reduced perplexity by a factor of 4 relative to the baseline. However, it merely slowed down the ASR by orders of magnitude without improving its accuracy.

E. Alignment methods

A key step in scoring oral reading is aligning the ASR output and manual transcript against the text to compute traces. The standard NIST align procedure is ill-suited to this purpose because it treats regressions (rereading one or more words) as insertions instead of as normal reading. Instead, we developed the MultiMatch alignment procedure to take regression into account.

MultiMatch uses dynamic programming to find the lowest-cost mapping from a sequence of recognized or transcribed words to positions in a text sentence. It imposes a mismatch penalty for aligning a word against a text word it does not match. This penalty reflects the orthographic and phonemic distance between them. MultiMatch imposes a jump penalty for a transition from position i to any position except i or i+1.

The penalties are set to prefer an isolated mismatch to jumping to a word and back. For instance, in aligning the reading *once upon the time …* to the text "Once upon a time the beautiful princess …," MultiMatch aligns *the* against the text word "a" rather than jump forward in the sentence to match the word "the" and back to match the word "time."

MultiMatch outputs alignments in both text space and speech space. The text space alignment associates each

word of text with at most one spoken word. The speech space alignment associates each recognized or transcribed word with the text word it is aligned against.

Having recognized the crucial importance of tracking accuracy and spending years trying to improve it, with scant success, we decided to address the problem of tracking by redefining it. We had framed this problem as "chasing the kid" – that is, finding whichever word the child was trying to read. We decided to reduce the problem to "blaming the kid" – that is, deciding whether the child was reading whichever word the tutor determined should come next, namely the earliest uncredited word in the sentence. The tutor could then simply wait to hear this word. To avoid getting stuck at false alarms, the tutor could skip over at most one text word to accept the next word. To make its behavior more understandable, the tutor could highlight the word it is waiting to hear (though it does not yet do so).

Sure enough, tracking accuracy was substantially higher with this redefined criterion [16]. However, when we modified the language model to use the same criterion, tracking accuracy suffered. We concluded that "chase the kid" was more accurate at tracking the child's actual position, even if we used "blame the kid" to indicate which word to read next. We believe the reason is that "chase the kid" is a more accurate model of actual reading behavior, and therefore tracks the reader's actual position more accurately. In contrast, the monotonic left-to-right "blame the kid" language model is apt to get lost when it fails to follow the reader. The lesson is to use a faithful model of reading to track the reader's actual position, even if the tutor refrains from displaying it externally. I.e., ***rely on realism*** but ***mask mistracking.***

F. Prosodic models

Expressiveness is an important aspect of oral reading fluency. To assess children's oral reading fluency, we built on work [60-65] by Schwanenflugel and her colleagues, who analyzed the development of children's oral reading prosody and related it to their gains in fluency and comprehension. Given a child's oral reading of a sentence, they measured its expressiveness by correlating its prosodic contour – that is, the word-by-word sequence of pitch, duration, and intensity – against adult prosodic contours for the same sentence.

First we scaled up from Schwanenflugel *et al.*'s painstakingly hand-measured prosodic features of a few utterances to comprehensive automated assessments of children's prosodic contours by correlating them against the contours of the Reading Tutor's recorded fluent adult narrations of the same sentences [66]. We analyzed the sensitivity of this template-based measure to prosodic improvements in a child's successive readings of a sentence on the same or different days [67].

Then we generalized this approach by using the adult narrations to train a normative model of oral reading prosody, and using the trained model to score children's oral reading prosody [68]. The generalized model outperformed the template-based measure in predicting children's end-of-year scores and gains in fluency and comprehension [21]. It used only duration information, but latencies are very informative. That is, ***silences are golden.***

Next we used the generalized model to mine a corpus of children's oral reading in order to identify the specific common syntactic and lexical features of text on which children scored best and worst. These features predicted their fluency and comprehension test scores and gains better than the previous models.

Meanwhile, to explore how to give children real-time feedback on their oral reading prosody, we developed a flexible prosody visualization tool for mapping each word's prosodic features to graphical features, in order to user-test experimenter-specified mappings [8]. For instance, this tool can map a word's pitch to its vertical position, loudness to font size, and temporal features to the timing of the dynamic display. It can map multiple features to different dimensions of color, such as hue, saturation, and intensity. Mapping "adult-likeness" to hue provides visual feedback on the proximity of the child's pitch, duration, and/or intensity to the narrator's. Mapping latency to intensity makes higher-latency words pale so as to reflect tentative, hesitant reading. Mapping ASR confidence to saturation makes lower-confidence words look more like unread words, to reflect uncertainty that they were read correctly.

V. CONCLUSIONS

In over two decades of applying speech recognition to children's oral reading, Project LISTEN has learned a number of lessons about what worked – and more often, what didn't, at least for us – and found some hard questions:

Acoustic models: ***Quality trumps quantity.*** Augmenting a large corpus of manually transcribed oral reading with ASR output filtered to serve as automated transcripts hurt accuracy. Is there a way to make it help?

Lexical models: ***Distracters detract.*** Except for phonetic truncations of sentence words, predicting likely miscues detected more of them only by hallucinating them as well. What if any distracters are worth listening for?

Confidence scores: ***Tracking trips up scoring.*** Confidence scores of mistracked words are useless. How can confidence scores be made robust to mistracking?

Language models: ***Rely on realism.*** The better we model children's oral reading, the better we can track it. What if any models boost tracking accuracy dramatically?

Alignment models: ***Mask mistracking.*** It's easier to tell if children are at the right spot than where they are instead, and even easier to prompt them to click but not say where. Can alignment plus interface redesign hide tracking errors?

Prosodic models: ***Silences are golden.*** Duration of latency between words is a good gauge of reading fluency. How much can tracking better make latency measure better?

ASR is notoriously empirical, so what failed for us may work for others, and possibly vice versa. Thus these lessons come without guarantees of generality. However, if they steer readers towards fruitful approaches and away from fruitless ones, they will have served a useful purpose.

REFERENCES (many at www.cs.cmu.edu/~listen)

[1] M. J. Adams, "The promise of automatic speech recognition for fostering literacy growth in children and adults," in *International Handbook of Literacy and Technology*. vol. 2, M. McKenna, L. Labbo, R. Kieffer, and D. Reinking, Eds. Hillsdale, NJ: Lawrence Erlbaum Associates, 2006, pp. 109-128.

[2] J. Mostow and G. S. Aist, "Giving help and praise in a reading tutor with imperfect listening -- because automated speech recognition means never being able to say you're certain," *CALICO Journal*, vol. 16, pp. 407-424, 1999.

[3] A. Hagen, B. Pellom, and R. Cole, "Highly accurate children's speech recognition for interactive reading tutors using subword units," *Speech Communication*, vol. 49, pp. 861-873, December 2007.

[4] V. L. Beattie, "Scientific Learning Reading Assistant™: CMU Sphinx technology in a commercial educational software application," in *CMU Sphinx Users and Developers Workshop*, Dallas, TX, 2010.

[5] W. Chen, J. Mostow, and G. Aist, "Using Automatic Question Generation to Evaluate Questions Generated by Children," in *Proceedings of the AAAI Symposium on Question Generation*, Arlington, VA, 2011.

[6] W. Chen and J. Mostow, "A Tale of Two Tasks: Detecting Children's Off-Task Speech in a Reading Tutor," in *Interspeech: Proceedings of the 12th Annual Conference of the International Speech Communication Association*, Florence, Italy, 2011.

[7] X. Zhang, J. Mostow, N. K. Duke, C. Trotochaud, J. Valeri, and A. Corbett, "Mining Free-form Spoken Responses to Tutor Prompts," in *Proceedings of the First International Conference on Educational Data Mining*, Montreal, 2008, pp. 234--241.

[8] S. Sitaram, J. Mostow, Y. Li, A. Weinstein, D. Yen, and J. Valeri, "What visual feedback should a reading tutor give children on their oral reading prosody?," in *Proceedings of the Third ISCA Workshop on Speech and Language Technology in Education (SLaTE)*, Venice, Italy, 2011.

[9] G. Aist and J. Mostow, "A time to be silent and a time to speak: Time-sensitive communicative actions in a reading tutor that listens," in *AAAI Fall Symposium on Communicative Actions in Humans and Machines*, Boston, MA, 1997.

[10] G. Aist, "Expanding a time-sensitive conversational architecture for turn-taking to handle content-driven interruption," in *Proceedings of the International Conference on Speech and Language Processing (ICSLP98)*, Sydney, Australia, 1998, p. #928.

[11] J. Mostow and G. Aist, "The sounds of silence: Towards automated evaluation of student learning in a Reading Tutor that listens," in *Proceedings of the Fourteenth National Conference on Artificial Intelligence (AAAI-97)*, Providence, RI, 1997, pp. 355-361.

[12] J. E. Beck, P. Jia, and J. Mostow, "Automatically assessing oral reading fluency in a computer tutor that listens," *Technology, Instruction, Cognition and Learning*, vol. 2, pp. 61-81, 2004.

[13] J. Mostow, "Is ASR accurate enough for automated reading tutors, and how can we tell?," in *Proceedings of the Ninth International Conference on Spoken Language Processing (Interspeech 2006 — ICSLP), Special Session on Speech and Language in Education*, Pittsburgh, PA, 2006, pp. 837-840.

[14] J. Mostow, A. G. Hauptmann, L. L. Chase, and S. Roth, "Towards a reading coach that listens: automated detection of oral reading errors," in *Proceedings of the Eleventh National Conference on Artificial Intelligence (AAAI93)*, Washington, DC, 1993, pp. 392-397.

[15] G. Doddington, "Word Alignment Issues in ASR Scoring," in *IEEE Workshop on Automatic Speech Recognition and Understanding (ASRU '03)*, St. Thomas, U.S. Virgin. Islands, 2003, pp. 630- 633.

[16] M. H. Rasmussen, J. Mostow, Z.-H. Tan, B. Lindberg, and Y. Li, "Evaluating Tracking Accuracy of an Automatic Reading Tutor," in *Proceedings of the Third ISCA Workshop on Speech and Language Technology in Education (SLaTE)*, Venice, Italy, 2011.

[17] Y. Li and J. Mostow, "Evaluating and improving real-time tracking of children's oral reading," in *Proceedings of the 25th Florida Artificial Intelligence Research Society Conference (FLAIRS-25)*, Marco Island, Florida, 2012.

[18] J. E. Beck, P. Jia, J. Sison, and J. Mostow, "Predicting student help-request behavior in an intelligent tutor for reading," in *Proceedings of the 9th International Conference on User Modeling*, Johnstown, PA, 2003, pp. 303-312.

[19] X. Zhang, J. Mostow, and J. E. Beck, "Can a computer listen for fluctuations in reading comprehension?," in *Proceedings of the 13th International Conference on Artificial Intelligence in Education*, Marina del Rey, CA, 2007, pp. 495-502.

[20] J. E. Beck and J. Sison, "Using knowledge tracing in a noisy environment to measure student reading proficiencies," *International Journal of Artificial Intelligence in Education (Special Issue "Best of ITS 2004")*, vol. 16, pp. 129-143, 2006.

[21] M. Duong, J. Mostow, and S. Sitaram, "Two Methods for Assessing Oral Reading Prosody " *ACM Transactions on Speech and Language Processing (Special Issue on Speech and Language Processing of Children's Speech for Child-machine Interaction Applications)*, vol. 7, pp. 14:1-22, August 2011.

[22] S. Sitaram and J. Mostow, "Mining Data from Project LISTEN's Reading Tutor to Analyze Development of Children's Oral Reading Prosody," in *Proceedings of the 25th Florida Artificial Intelligence Research Society Conference (FLAIRS-25)*, Marco Island, Florida, 2012.

[23] P. Jia, J. E. Beck, and J. Mostow, "Can a Reading Tutor that Listens use Inter-word Latency to Assess a Student's Reading Ability?," in *Proceedings of the ITS 2002 Workshop on Creating Valid Diagnostic Assessments*, San Sebastian, Spain, 2002, pp. 23-32.

[24] G. Aist and J. Mostow, "Using Automated Within-Subject Invisible Experiments to Test the Effectiveness of Automated Vocabulary Assistance," in *Proceedings of ITS'2000 Workshop on Applying Machine Learning to ITS Design/Construction*, Montreal, Canada, 2000, pp. 4-8.

[25] J. Mostow, "Experience from a Reading Tutor that listens: Evaluation purposes, excuses, and methods," in *Interactive literacy education: facilitating literacy environments through technology*, C. K. Kinzer and L. Verhoeven, Eds. New York: Lawrence Erlbaum Associates, Taylor & Francis Group, 2008, pp. 117-148.

[26] C. Heiner, J. E. Beck, and J. Mostow, "Improving the help selection policy in a Reading Tutor that listens," presented at the Proceedings of the InSTIL/ICALL Symposium on Natural Language Processing and Speech Technologies in Advanced Language Learning Systems, Venice, Italy, 2004.

[27] C. Heiner, J. E. Beck, and J. Mostow, "When do students interrupt help? Effects of time, help type, and individual differences," in *Proceedings of the 12th International Conference on Artificial Intelligence in Education (AIED 2005)*, Amsterdam, 2005, pp. 819-826.

[28] K.-m. Chang, J. Beck, J. Mostow, and A. Corbett, "A Bayes Net Toolkit for Student Modeling in Intelligent Tutoring Systems," presented at the Proceedings of the 8th International Conference on Intelligent Tutoring Systems, Jhongli, Taiwan, 2006.

[29] J. E. Beck, K.-m. Chang, J. Mostow, and A. Corbett, "Does help help? Introducing the Bayesian Evaluation and Assessment methodology," in *9th International Conference on Intelligent Tutoring Systems*, Montreal, 2008, pp. 383-394. ITS2008 Best Paper Award.

[30] J. E. Beck and J. Mostow, "How who should practice: Using learning decomposition to evaluate the efficacy of different types of practice for different types of students [Best Paper Nominee]," in *9th International Conference on Intelligent Tutoring Systems*, Montreal, 2008, pp. 353-362.

[31] J. E. Beck, "Using learning decomposition to analyze student fluency development," in *ITS2006 Educational Data Mining Workshop*, Jhongli, Taiwan, 2006, pp. 21-28.

[32] J. E. Beck, "Does learner control affect learning?," in *Proceedings of the 13th International Conference on Artificial Intelligence in Education*, Los Angeles, CA, 2007, pp. 135-142.

[33] X. Zhang, J. Mostow, and J. E. Beck, "All in the (word) family: Using learning decomposition to estimate transfer between skills in a

Reading Tutor that listens," in *AIED2007 Educational Data Mining Workshop*, Marina del Rey, CA, 2007.

[34] J. M. Leszczenski, "Learning Factors Analysis Learns to Read," Masters Thesis, School of Computer Science, Carnegie Mellon University, Pittsburgh, PA, 2007.

[35] J. Mostow, G. Aist, C. Huang, B. Junker, R. Kennedy, H. Lan, D. Latimer, R. O'Connor, R. Tassone, B. Tobin, and A. Wierman, "4-Month evaluation of a learner-controlled Reading Tutor that listens," in *The Path of Speech Technologies in Computer Assisted Language Learning: From Research Toward Practice*, V. M. Holland and F. P. Fisher, Eds. New York: Routledge, 2008, pp. 201-219.

[36] J. Mostow, G. Aist, P. Burkhead, A. Corbett, A. Cuneo, S. Eitelman, C. Huang, B. Junker, M. B. Sklar, and B. Tobin, "Evaluation of an automated Reading Tutor that listens: Comparison to human tutoring and classroom instruction," *Journal of Educational Computing Research*, vol. 29, pp. 61-117, December 2003.

[37] J. Mostow, J. Nelson, and J. Beck, "Computer-Guided Oral Reading versus Independent Practice: Comparison of Sustained Silent Reading to an Automated Reading Tutor that Listens," *Journal of Educational Psychology*, under review.

[38] R. Poulsen, P. Wiemer-Hastings, and D. Allbritton, "Tutoring Bilingual Students with an Automated Reading Tutor That Listens," *Journal of Educational Computing Research*, vol. 36, pp. 191-221, 2007.

[39] G. A. Korsah, J. Mostow, M. B. Dias, T. M. Sweet, S. M. Belousov, M. F. Dias, and H. Gong, "Improving Child Literacy in Africa: Experiments with an Automated Reading Tutor," *Information Technologies and International Development*, vol. 6, pp. 1-19, 2010.

[40] F. Weber and K. Bali, "Enhancing ESL Education in India with a Reading Tutor that Listens," presented at the Proceedings of the First ACM Symposium on Computing for Development London, United Kingdom, 2010.

[41] K. Reeder, J. Shapiro, and J. Wakefield, "A computer based reading tutor for young English language learners: recent research on proficiency gains and affective response," in *16th European Conference on Reading and 1st Ibero-American Forum on Literacies*, University of Minho, Campus de Gualtar, Braga, Portugal, 2009.

[42] T. Cunningham, "The Effect of Reading Remediation Software on the Language and Literacy Skill Development of ESL Students," Master's thesis, Department of Human Development and Applied Psychology, University of Toronto, Toronto, Canada, 2006.

[43] J. Cohen, *Statistical power analysis for the behavioral sciences*, 2nd ed. Hillsdale, NJ: Lawrence Erlbaum Associates, 1988.

[44] CMU, "The CMU Sphinx Group open source speech recognition engines [software at http://cmusphinx.sourceforge.net]," ed, 2008.

[45] A. G. Hauptmann, L. L. Chase, and J. Mostow, "Speech Recognition Applied to Reading Assistance for Children: A Baseline Language Model," in *Proceedings of the 3rd European Conference on Speech Communication and Technology (EUROSPEECH93)*, Berlin, 1993, pp. 2255-2258.

[46] J. Mostow, S. F. Roth, A. G. Hauptmann, and M. Kane, "A prototype reading coach that listens [AAAI-94 Outstanding Paper]," in *Proceedings of the Twelfth National Conference on Artificial Intelligence*, Seattle, WA, 1994, pp. 785-792.

[47] G. Aist, P. Chan, X. D. Huang, L. Jiang, R. Kennedy, D. Latimer, J. Mostow, and C. Yeung, "How effective is unsupervised data collection for children's speech recognition?," in *Proceedings of the International Conference on Speech and Language Processing (ICSLP98)*, Sydney, Australia, 1998, p. #929.

[48] M. Eskenazi, "KIDS: A database of children's speech," *Journal of the Acoustic Society of America*, vol. 100, p. 2, December 1996 1996.

[49] M. Eskenazi and J. Mostow, "The CMU KIDS Speech Corpus (LDC97S63)," ed: Linguistic Data Consortium (http://www.ldc.upenn.edu), University of Pennsylvania, 1997.

[50] CMU. *The CMU Pronouncing Dictionary*. Available: http://www.speech.cs.cmu.edu/cgi-bin/cmudict

[51] E. A. Betts, *Foundations of Reading Instruction*. New York: American Book Company, 1946.

[52] J. Fogarty, L. Dabbish, D. Steck, and J. Mostow, "Mining a database of reading mistakes: For what should an automated Reading Tutor listen?," in *Artificial Intelligence in Education: AI-ED in the Wired and Wireless Future*, J. D. Moore, C. L. Redfield, and W. L. Johnson, Eds. San Antonio, Texas: Amsterdam: IOS Press, 2001, pp. 422-433.

[53] J. Mostow, J. Beck, S. V. Winter, S. Wang, and B. Tobin, "Predicting oral reading miscues," in *Proceedings of the Seventh International Conference on Spoken Language Processing (ICSLP-02)*, Denver, CO, 2002, pp. 1221-1224.

[54] S. Banerjee, J. E. Beck, and J. Mostow, "Evaluating the effect of predicting oral reading miscues," in *Proc. 8th European Conference on Speech Communication and Technology (Eurospeech 2003)*, Geneva, Switzerland, 2003, pp. 3165-3168.

[55] Y.-C. Tam, J. Mostow, J. Beck, and S. Banerjee, "Training a Confidence Measure for a Reading Tutor that Listens," in *Proc. 8th European Conference on Speech Communication and Technology (Eurospeech 2003)*, Geneva, Switzerland, 2003, pp. 3161-3164.

[56] M. Ravishankar, R. Bisiani, and E. Thayer, "Sub-Vector Clustering to Improve Memory and Speed Performance of Acoustic Likelihood Computation," in *Proceedings of Eurospeech*, Rhodes, Greece, 1997, pp. 151-154.

[57] D. Bansal and M. Ravishankar, "New Features for Confidence Annotation," in *Proceedings of International Conference on Spoken Language Processing (ICSLP)*, Sydney, Australia, 1998.

[58] S. Banerjee, J. Mostow, J. E. Beck, and W. Tam, "Improving Language Models by Learning from Speech Recognition Errors in a Reading Tutor that Listens," in *Second International Conference on Applied Artificial Intelligence*, Fort Panhala, Kolhapur, India, 2003, pp. 187-193.

[59] T. M. Mitchell, P. E. Utgoff, and R. B. Banerji, "Learning by experimentation: acquiring and refining problem-solving heuristics," in *Machine Learning*, R. S. Michalski, J. G. Carbonell, and T. M. Mitchell, Eds. Palo Alto, CA: Tioga, 1983, pp. 163-190.

[60] P. J. Schwanenflugel, A. M. Hamilton, M. R. Kuhn, J. M. Wisenbaker, and S. A. Stahl, "Becoming a fluent reader: Reading skill and prosodic features in the oral reading of young readers," *Journal of Educational Psychology*, vol. 96, pp. 119-129, 2004.

[61] J. Miller and P. J. Schwanenflugel, "Prosody of syntactically complex sentences in the oral reading of young children," *Journal of Educational Psychology*, vol. 98, pp. 839-853, 2006.

[62] P. J. Schwanenflugel, M. R. Kuhn, R. D. Morris, and B. A. Bradley. (2006, November 8). *The Development of Fluent and Automatic Reading: Precursor to Learning from Text*. Available: http://drdc.uchicago.edu/community/project.phtml?projectID=60

[63] J. Miller and P. J. Schwanenflugel, "A longitudinal study of the development of reading prosody as a dimension of oral reading fluency in early elementary school children," *Reading Research Quarterly*, vol. 43, pp. 336–354, 2008.

[64] R. G. Benjamin and P. J. Schwanenflugel, "Text complexity and oral reading prosody in young readers," *Reading Research Quarterly*, vol. 45, pp. 388-404, October/November/December 2010.

[65] M. R. Kuhn, P. J. Schwanenflugel, and E. B. Meisinger, "Aligning Theory and Assessment of Reading Fluency: Automaticity, Prosody, and Definitions of Fluency. Invited review article," *Reading Research Quarterly*, vol. 45, pp. 230–251, Apr-Jun 2010.

[66] J. Mostow and M. Duong, "Automated Assessment of Oral Reading Prosody," in *Proceedings of the 14th International Conference on Artificial Intelligence in Education (AIED2009)*, Brighton, UK, 2009, pp. 189-196.

[67] M. Duong and J. Mostow, "Detecting prosody improvement in oral rereading," in *Online Proceedings of the Second ISCA Workshop on Speech and Language Technology in Education (SLaTE)*, Wroxall Abbey Estate, Warwickshire, England, 2009, p. at http://www.eee.bham.ac.uk/SLaTE2009/.

[68] M. Duong and J. Mostow, "Adapting a Duration Synthesis Model to Score Children's Oral Reading," in *Interspeech 2010*, Makuhari, Japan, 2010, pp. 769-772.

Adaptive and Discriminative Modeling for Improved Mispronunciation Detection

Horacio Franco, Luciana Ferrer, and Harry Bratt
Speech Technology and Research Laboratory
SRI International
Menlo Park, CA, USA

***Abstract*—In the context of computer-aided language learning, automatic detection of specific phone mispronunciations by nonnative speakers can be used to provide detailed feedback for a student to work with specific pronunciation problems when producing the new sounds of a foreign language. Starting with an initial approach based on a measure of match to native models, we found that significant improvements could be achieved by explicitly modeling both mispronunciations and correct pronunciations by nonnative speakers. In this work, our approach is extended based on the use of adaptation and discriminative modeling, showing significant improvements from our previous best system. Performance of the proposed approaches was evaluated in a phonetically transcribed database of 130,000 phones uttered in continuous speech sentences by 206 nonnative speakers.**

Mispronunciation detection; pronunciation scoring; computer-aided language learning

I. INTRODUCTION

Using computers to help students learn and practice a new language has long been seen as a promising area for the use of automatic speech recognition (ASR) technology. It could allow spoken language to be used in many ways in language-learning activities, for example by supporting different types of oral practice and enabling feedback on various dimensions of language proficiency, including language use and pronunciation quality. A desirable feature of the use of speech technology for computer-aided language learning (CALL) is the ability to provide meaningful feedback on pronunciation quality.

Nevertheless, current speech recognition technology has limitations on accuracy, and designers of CALL systems that use ASR must design the applications to minimize the impact of such limitations. In particular, in the area of pronunciation scoring, the smaller the unit to be scored, the higher the uncertainty in the associated score [1]. Currently, the most reliable estimates of pronunciation quality are overall levels obtained from a paragraph composed of several sentences that can be used to characterize the speaker's overall pronunciation proficiency. At this level, it has been shown that automatic scoring performs as well as human scoring [2].

For many CALL applications we would like to score smaller units, to allow the student to focus on specific aspects of that student's own speech production. For instance, overall pronunciation scoring can be obtained at the sentence level [3], [4], with a level of accuracy that, while lower than that of human scoring, can nonetheless provide valuable feedback for language learning [5]. However, an overall score is only part of the desired feedback for pronunciation training. More detailed feedback, at the level of individual phones, directs attention to specific phones that are mispronounced [1], [6-11]. At this level of detail, typically only a binary decision between correct and mispronounced is provided.

In earlier work [9], we compared two approaches for phone mispronunciation detection. The first was based on using native models as a reference, and on obtaining a measure of the phone-level degree of match to a corresponding native model [1]. The second approach was based on using explicit acoustic models for the correct and for the mispronounced utterances of a phone, and on computing a likelihood ratio using these two models. We found that the second approach was more accurate, resulting in an average 9% relative reduction in the equal error rate (EER) of the mispronunciation detection.

In this paper we extend the work on acoustic modeling of correct and mispronounced nonnative phones by using more advanced acoustic modeling techniques based on adaptation and discriminative modeling. The proposed techniques are inspired by approaches that have been effective on significantly improving accuracy in another area of speech technology – namely, speaker recognition. The first proposed approach uses model adaptation in a form that is inspired by the Gaussian Mixture Model-Universal Background Model (GMM-UBM) speaker verification/detection system proposed by Reynolds et al. [12]. This approach has shown to be more effective than other adaptation approaches when used for a detection task, particularly when limited adaptation data is available. The second proposed approach is based on the use of discriminant classifiers based on support vector machines (SVMs) using as an input feature a GMM supervector consisting of the stacked means and weights of the mixture components. The GMM supervector is obtained by adapting a GMM-UBM to a test utterance [13]. We also explored the combination of the scores from these two approaches.

The outline of this paper is as follows: in Section 2 we review the baseline approach and present the new approaches explored in this paper, in Section 3 we describe the database that we use to evaluate them, and in Section 4 we present our experimental results comparing the different approaches. Section 5 concludes this work.

II. PHONE-LEVEL MISPRONUNCIATION DETECTION APPROACHES

Earlier modeling approaches [1], [7] used basically a native model to produce a measure of goodness for the nonnative speech. While this measure correlates very well with human judgments for longer segments (i.e., paragraphs or sentences), the correlation decreases for shorter segments, such as phones [1]. One possible explanation is that the human judgments of pronunciation are less consistent on shorter segments. Also, the available acoustic information is much less on a phone segment than in a sentence or paragraph, and we cannot take advantage of averaging, possibly noisy, acoustic scores over many acoustic frames to obtain more reliable scores. Another possibility is that measures derived based on the native model are not accurate enough to capture consistently the differences in nonnative pronunciations at the phone level. To address this issue, an alternative approach [8] used hidden Markov models (HMMs) with two alternative pronunciation models per phone – one trained on native speech and the other on strongly accented nonnative speech. Mispronunciations were detected from the phone backtrace. Performance was limited by the fact that there was no training data with specific phone-level mispronunciation information. After collecting such data [14], in [9] we explored more detailed acoustic modeling to attempt to capture the subtle differences between the nonnative speech realizations that are considered acceptable versus the nonnative speech realizations that are considered mispronounced. In that work we compared two mispronunciation detection schemes. The first approach was based on phone log-posterior scores [15], [1]. The phone log-posterior scores are similar to the GOP scores introduced in [7], but log-posterior probabilities are computed at the frame level, and averaged over the frames of a given phone segment. The second approach was based on explicit acoustic modeling, using GMMs, of the nonnative productions of correct and mispronounced phones [9]. A log-likelihood ratio (LLR) of mispronounced and correct phone models was used as the measure of pronunciation quality in the second method. We did find significant gains from the use of the explicit mispronunciation modeling using the GMMs LLR approach. Nevertheless, it should be noted that the explicit acoustic modeling of mispronunciation comes at a price: it is necessary to collect and annotate a nonnative training database, there is typically less data to train the corresponding nonnative acoustic modes, and the resulting models are dependent on the first language of the nonnative speakers.

In this work we aim to further develop the explicit modeling of mispronunciations by using newer acoustic modeling techniques that can be more effective dealing with the challenges of this task. In the following approaches we assumed that the phonetic segmentation is known, typically obtained by computing a forced alignment with an HMM, using the speech transcription and the pronunciation dictionary.

A. System 1: LLR of independently trained GMMs

Our baseline approach for mispronunciation detection is the GMM LLR proposed in [9], where for each phone class we trained two different GMMs: one model is trained with the "correct" native-like pronunciations of a phone, while the other model is trained with the "mispronounced" or nonnative pronunciations of the same phone. In the evaluation phase, for each phone segment q_i, a length-normalized log-likelihood ratio score $LLR(q_i)$ was computed by using the "mispronounced", λ_M, and the "correct", λ_C, pronunciation models, respectively, where $LLR(q_i)$ is defined in eq. (1).

$$LLR(q_i) = \frac{1}{d_i} \sum_{t=t_i}^{t_i+d_i-1} [\log P(y_t \mid q_i, \lambda_M) - \log P(y_t \mid q_i, \lambda_C)] \quad (1)$$

where $P(y_t \mid \overline{q_i}, \lambda)$ is the probability of the acoustic feature y_t for the frame at time t given the phone class q_i and the model λ. The normalization by the phone duration d_i allows definition of unique thresholds for the LLR for each phone class, independent of the lengths of the segments. A mispronunciation is detected when the LLR is above a predetermined threshold, specific to each phone. In this system we used diagonal covariance GMMs for all phone models, and for each phone and each class ("correct" or "mispronounced"), the corresponding GMM was created with a number of mixture components proportional to the number of samples for the phone for that class. The proportion is a tunable parameter that can be optimized.

B. System 2: LLR of adapted GMMs

This system is similar to baseline System 1, except that the models for each class ("correct" and "mispronounced") for each phone are obtained by adaptation. The model to which they are adapted is trained using all the samples from a given phone, ignoring the class. We use Bayesian adaptation [16] to adapt this class-independent GMM to all the "correctly" pronounced training examples for a phone, and obtain in this way the adapted "correct" model for such phone. We also adapt the same class–independent GMM to the "mispronounced" training examples for the same phone, producing the adapted "mispronounced" model for such phone. For these class-dependent GMMs we adapt both the means and the mixture weights to the class-specific data.

With these two adapted models we can compute, for any test phone, the LLR of the adapted "mispronounced" model to the adapted "correct" model. The key point in this approach is that the models used to compute the LLR are not trained independently, but are derived from the same class-independent model. This provides a tighter coupling between the "mispronounced" and the "correct" model, which has been shown to produce better performance in the area of speaker detection [12].

C. System 3: SVM classifier based on adapted GMM supervector

Inspired by the work on [13], we use the class-independent GMM trained for System 2 to create supervectors by adapting this GMM to each phone instance. The supervector for a certain phone instance is obtained by adapting the means and mixture weights of the original GMM to the acoustic feature vector representing the phone. This operation corresponds to a

transformation of the phone segment acoustic feature vectors into a fixed high-dimension feature vector in the GMM supervector space. In our preliminary experimentation, we found that adapting only the means or only the weights of the GMM gives worse performance than adapting both means and weights. The supervectors are then normalized to have the same variance in all dimensions and fed to a linear SVM [17].

Consider a training set with m samples, $S = (x_i, z_i)$, $i = 1,...,m$, where x_i is the supervector and z_i is the class corresponding to sample i (-1 for a correct pronunciation, +1 for an incorrect one). The goal of the SVM is to find a function $f(x) = w^T x + b$, such that sign($f(x)$) is the predicted class for feature vector x. Parameters w and b are obtained by solving the following optimization problem:

$$\min J(w,\varepsilon) = \frac{1}{2} w^T w + C\left(\sum_{i:z_i=-1} \varepsilon_i + J \sum_{i:z_i=+1} \varepsilon_i \right) \qquad (2)$$

subject to $z_i(w^T x_i + b) \geq 1 - \varepsilon_i$, and $\varepsilon_i \geq 0$, for $i = 1,...,m$. The variables ε_i are called slack variables and measure the error incurred on each sample. In our experiments training parameters *C* (the regularization parameter) and *J* (the relative weight given to the error in the positive versus the negative samples) are obtained empirically.

Given a test phone instance, its supervector *x* is first computed (by adaptation of the class-independent GMM to its feature vector frames), and $f(x)$ (the distance of that supervector to the SVM hyperplane) is taken as the score for the sample.

D. System 4: Combination of Systems 2 and 3

We combined the two systems by using a simple weighted combination of the scores given by each of the two systems for each phone. No tuning was done on the weights because we lacked an additional development set with which to tune it. We used a weight of 0.25 for System 2 and 0.75 for System 3, since the range of the scores of System 3 is around three times smaller than the range for those of System 2. So, the weight was just used for equalizing the score ranges. The same weight was used for all phone classes for each system.

III. DATABASE DESCRIPTION

To evaluate these modeling approaches we need a nonnative speech database transcribed at the phone level on the pronunciation variants of interest. We used a phonetically transcribed subset of the nonnative Spanish database [14]. All speech data was read speech from Spanish newspapers with no repeated sentences, aiming at developing text-independent systems. Four native Spanish-speaking expert phoneticians transcribed 2550 sentences, totaling 130,000 phones, of nonnative speech data. Those sentences, randomly divided among the transcribers, were produced by 206 nonnative speakers whose native language was American English. Their levels of proficiency were varied, and an attempt was made to balance the number of speakers by level of proficiency as well as by gender. An additional set of sentences (one newspaper sentence from each of the 206 speakers), the common pool, was transcribed by all four phoneticians to assess the human-human consistency.

A first step in acquiring the phonetic transcriptions was to define the transcription conventions. As the native language of all the nonnative speakers was the same (American English), we could expect to observe a relatively small set of common pronunciation problems. Also, we were interested only in nonnative phones – phones that the transcribers perceived as natively produced did not need to be described in any detail. Given these issues, our approach was to define two sets of phones plus a set of diacritics. The first set of phones consisted of all the native phones in the targeted dialect of Spanish. The second set of phones consisted of phones of American English, such as some reduced vowels and the labio-dental fricative [v], which we expected to see carry over into nonnative pronunciations of Spanish. The diacritics were allowed to modify appropriate native phones. The transcribers were instructed that using a diacritic on a phone implied that the phone was not perceived as native, and the diacritic explained the way in which it was nonnative. Diacritics included aspiration for the voiceless stops, gliding for the nonlow vowels, and length (i.e., nonnatively long). A catch-all diacritic, "*", was included to represent a sound that was perceived as a nonnative rendition of a phone but for which no more specific method of indicating its nonnativeness was available. In this way, we reduced the transcription problem to a simpler one in terms of cognitive effort for the transcriber, and ease of information entry, while still encoding the most important piece of information in all the transcriptions – the judgment of the nativeness of any given phone. Furthermore, for this study, the detailed phone-level transcriptions were collapsed into two categories: native-like and nonnative pronunciations.

A. Human consistency in phone transcription

As the goal of the phone-level transcriptions is to train automatic systems to detect mispronunciations by nonnatives, before evaluating such systems we wanted to assess how consistently humans can perform this task. To that end we used the 206 common sentences from the transcribed database, and used the Kappa coefficient statistic [18], to determine how reliably the transcribers agree on the transcription for each of the 28 native phones. On nine of the phones, all four transcribers showed at least a moderate level of agreement (using $K > 0.41$ to mean "moderate" agreement). In Table 1 we show the value of Kappa for the phones with enough data in the common pool. In the third column we also show the percent of times that a phone is labeled as mispronounced by any of the transcribers working on the common sentences.

The most reliable phones to transcribe were the voiced stop approximants /β/, /ð/, and /γ/, which were also among the most frequently mispronounced. Phone /b/ was the most consistently transcribed, while flapped /r/, labial semivowel /w/, and palatal nasal /ñ/ had moderate correlations and also were frequently mispronounced. Phones /m/ and /s/ had high consistency among transcribers, but were not that frequently mispronounced. Vowel /i/ was the only vowel to have good

consistency. Some of the phones that were expected to be good predictors of nonnativeness, such as voiceless stops, most vowels, and /l/ and trilled /rr/, did not have good consistency across the transcribers.

TABLE I. CONSISTENCY AMONG THE TRANSCRIBERS IN LABELING THE PHONES, EVALUATED USING KAPPA FOR PHONES IN THE COMMON POOL. THE PERCENT OF TIMES THAT A PHONE IS LABELED AS MISPRONOUNCED IS SHOWN IN THE THIRD COLUMN. THE PHONES LABELED IN BOLD HAVE KAPPA > .41.

Phone	Kappa	% nonnative
a	0.26	17.
b	0.90	42.
β	0.70	73.
δ	0.55	68.
e	0.18	24.
γ	0.51	73.
i	0.41	20.
k	0.32	47.
l	0.22	28.
m	0.76	17.
n	0.15	6.
ñ	0.46	33.
o	0.23	20.
p	0.36	38.
r	0.36	42.
rr	0.29	77.
s	0.57	6.
t	0.34	34.
u	0.14	18.
w	0.43	40.
y	0.39	18.
z	0.35	84.

IV. EXPERIMENTAL SETUP AND RESULTS

Our mispronunciation detection approaches assume that the phonetic segmentation is given and accurate. Therefore, the task for which the mispronunciation detection is used must be designed to ensure a high speech recognition rate. Examples of such tasks are reading aloud and multiple-choice exercises.

For our experiments we generated phonetic alignments using the EduSpeak [19] HMM-based speech recognizer. The acoustic features were standard 13-dimensional Mel frequency cepstral coefficients (MFCCs) plus their delta coefficients obtained every 10 ms, based on a sliding 25-ms Hamming window. The C0 coefficient was normalized by the maximum over each sentence. Cepstral mean normalization was applied at the sentence level for C1 to C12. The acoustic models used to generate the phonetic alignments were gender independent, Genonic GMMs, as introduced in [20].

Given the alignments, the detection of mispronunciation is reduced to a binary classification of the phone's feature vectors, as the phone class is given by the alignments. Consequently, the mispronunciation detection performance can be studied independently for each phone class. Each of the mispronunciation detection approaches proposed produces a detection score. The performance of the mispronunciation detection algorithms was evaluated as a function of the threshold applied to that score, for each phone class. For each threshold we obtained the machine-produced labels "correct" (C) or "mispronounced" (M), for each phone utterance. Then, we compared the machine labels with the labels obtained from the phoneticians' transcriptions. For each threshold we computed two error measures: the probability of a false positive, estimated as the percent of cases where a phone utterance is labeled by the machine as incorrect when it was in fact correct, and the probability of a false negative – that is, the probability that the machine labeled a phone utterance as correct when it was in fact incorrect.

We evaluated the mispronunciation detection performance by computing the receiver operating characteristic (ROC) curve, and finding the points of equal error rate (EER), where the probability of false positive is equal to the probability of false negative. This error measure is independent of the actual priors for the C or M labels, but results in a higher total error rate than the possible minimum when the priors are skewed. Note that in an actual application of the mispronunciation detection system, criteria other than the EER may define an operating point along the ROC curve. For instance, for pedagogical reasons we may want to impose a maximum acceptable level of false positives.

To maximize the use of the phonetically transcribed data in evaluating the four proposed systems, we used a four-way jackknifing procedure to train and test these approaches on the same phonetically transcribed nonnative database. We trained models using data from three partitions and tested on the remaining partition, rotating the procedure four times over the four partitions. There were no common speakers across any of the partitions. When reporting results for a given system, the errors obtained for each of the four partitions were pooled to obtain mispronunciation detection average performance on the complete database.

A. System training and tuning

We define experimental details on the training of the proposed systems.

For System 1 we replicated with our current GMM training software the results from the original work [9]. As different phones have different amounts of training data, we explored the number of mixture components to use for each phone class GMM, and found that the proportion of 25 training samples for each mixture component resulted in the best performance for this system. The number of mixture components ranged from 2 to 531.

In developing System 2 we found that the optimal number of mixture components of the class-independent GMM for each phone was approximately the sum of the sizes of the class-dependent GMMs for each phone from System 1. We estimated the Bayesian adapted model using the detailed procedure described in [12]. For the adapted parameters, we explored a range of values for the relevance factor r, (16, 4, 1, 0), and found that the best results were obtained for the two lower values of r, meaning that little or no weight was given to the class-independent GMM parameters. Still, the class-independent GMM is used to compute the posterior

probabilities and the sufficient statistics for the means and the weights of each mixture component, and to provide the structure and the variance parameters (that were not adapted) for the class-adapted models, imposing in this way some consistency across the class-adapted models.

To train the SVM for System 3 we used the software package SVM-light [17], which can efficiently handle large training sets and allows for asymmetric cost factors. Optimal value for C (Equation 2) was the default one, computed as the average of the inverse of the norms of the input vectors. Optimal value for J was given by the ratio of the number of negative and positive samples available for training.

B. Experimental results

We show experimental results for all systems in Table 2 and in Figure 1. For all the systems and for every phone we computed the ROC curve and obtained the EER for mispronunciation detection for that phone. For each system, we also show weighted averages of the EER over all phones, where the weight was the relative frequency of each phone.

The independent GMM approach of System 1, the baseline system, had an average EER of 31.8%. The phones with the best performance were the approximants /β/,/ð/,/ɣ/, the voiced stop /b/, the semivowel /w/, the nasal /m/, and the fricative /x/. These phones have very good agreement with the phones with the highest Kappa in Table 1.

The use of adaptation in System 2 produced a significant reduction of EER across most of the phones. The largest improvements occurred for the voiced stop /g/, the voiceless velar fricative /x/, the voiceless palatal affricate /ʧ/ and /ɣ/, (21.6%, 11.6%, 10.7%, and 10.1% relative reduction with respect to System 1, respectively). These phone classes had less data than the average, showing that the adaptation approach is effective in dealing with smaller amounts of training data. Some of the largest EER reductions occurred for phones that were not necessarily among the most reliably transcribed, which suggests that the adaptation approach took advantage of additional useful information in the noisy transcriptions. The overall weighted EER reduction with the adaptation approach was 3.5% relative to the baseline system. There were also a few phones where the EER was slightly worse than the baseline, mainly for /s/, /d/, and /n/.

The SVM-based System 3 produced a bigger average EER reduction than System 2. Compared with System 1, the biggest EER reductions happened in phones /g/, / ñ /, /ʧ/, and /p/ (24%, 20%, 16%, and 14% relative EER reductions, respectively). Comparing Systems 2 and 3, we find that significant additional EER reductions occurred for the phones /ñ/ and /p/ (18.8% and 11.5% relative reductions, respectively), while some other phones had smaller gains. There was also significantly degraded performance for phones /x/, /b/, /ɣ/ (-29.4%, -11.7%, -11.2%) relative to System 2. These performance losses were unexpected, as these phones are among the set of best-performing phones in the baseline system, and they are also among the phones with high consistency in their transcriptions. They do have less training data compared to the other phones in the set of best-performing phones in the baseline system, which might have had an effect on this approach. The overall weighted EER reduction of System 3 with respect to System 1 and System 2 was 4.7% and 1.3%, respectively.

TABLE II. EQUAL ERROR RATE AT THE PHONE LEVEL FOR THE FOUR SYSTEMS STUDIED. WEIGHTED AVERAGES OF THE EER FOR EACH SYSTEM ARE SHOWN AT THE BOTTOM. WE ALSO SHOW THE NUMBER OF SAMPLES LABELED AS CORRECT OR MISPRONOUNCED FOR EACH PHONE.

	Equal Error Rate				# samples	
Phone	Sys. 1	Sys. 2	Sys. 3	Sys. 4	Corr.	Mispr.
b	13.29	12.61	14.08	11.83	668	490
x	14.22	12.57	16.27	12.70	752	190
m	14.43	14.54	14.19	13.85	4037	874
w	15.61	14.65	14.51	14.14	922	628
ð	17.67	17.50	17.59	16.50	1188	2497
ɣ	17.59	15.82	17.59	16.98	283	788
g	25.88	20.28	19.58	18.12	1124	143
β	21.25	19.66	20.05	18.42	554	1471
ñ	23.43	23.12	18.77	19.52	1157	559
z	22.25	22.37	22.70	21.39	238	1246
i	25.60	24.83	24.17	23.58	6153	1539
s	25.42	26.74	26.58	25.06	9520	587
l	28.33	27.77	26.72	26.09	4358	1729
t	31.13	29.28	27.85	28.13	3671	1902
p	32.98	31.84	28.17	28.25	2098	1310
k	32.25	29.60	30.53	28.80	2142	1861
y	32.30	30.82	30.94	29.06	3050	730
r	33.89	32.54	31.71	30.45	4561	3258
u	34.29	31.93	32.27	31.29	2420	592
rr	35.72	33.61	33.12	31.65	613	2192
a	34.62	32.74	33.62	32.41	12655	2617
ʧ	41.71	37.23	34.99	34.79	510	129
e	38.20	35.90	36.43	34.99	13258	4385
d	36.46	38.12	36.46	36.51	965	107
o	40.53	39.13	37.84	36.89	10072	2627
n	40.93	42.10	40.26	39.44	8944	601
Avg.	31.78	30.69	30.29	29.25	-	-

System 4, based on combining the scores of System 2 and System 3, was the best-performing system. It produced improvements over System 3 mostly on the phones /x/ and /b/ that had been degraded by System 3, also having moderate gains over most phones. Comparing System 4 with the baseline System 1 we appreciate that the combination of systems produces large gains on most phones ranging from relative EER reductions of 29.6% for /g/ to ~18% for /ʧ/, /n/, and /p/, and 10% to 12% for /β/, /rr/, /x/, /b/, /t/, /u/, /w/, gradually going down for the remaining phones. It is noteworthy that the EER reductions obtained by System 2 and System 3 with respect to the baseline resulted in reinforcing EER reductions for many phones in the combined system (see, for instance, /g/, /rr/, /ʧ/, etc.) Overall, the weighted EER relative reduction of System 4 with respect to the baseline System 1 was 8%, showing an almost additive combination of the average gains from System 2 and System 3. These results suggest that the improvements brought by these two systems over System 1 are highly complementary. Also, we observe that in a few cases

the EER of the combination system was slightly higher than either of the combined systems, suggesting that per-phone combination weights could improve System 4.

In general, the phones with the lowest EER were also those more consistently labeled by the transcribers. Nine phones had EER below 20%; among those, seven had Kappa above 0.41 (the other two did not have enough data in the common sentences data set to assess Kappa).

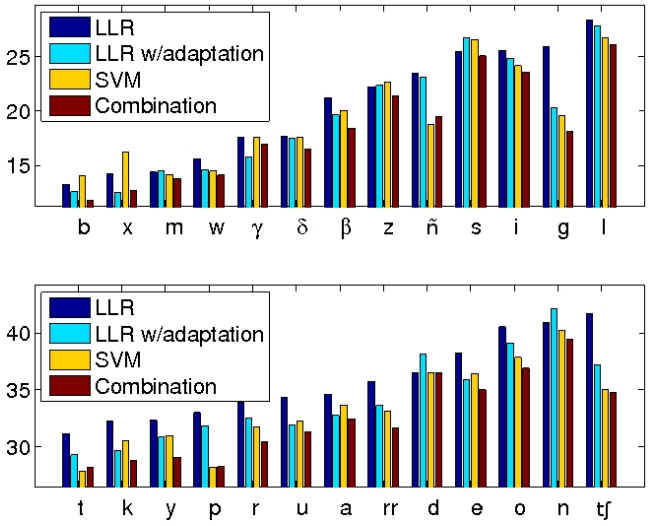


Figure 1. EER for the four systems for each phone in increasing order of EER for the baseline System 1 (LLR). System 2 is denoted as LLR w/adaptation, System 3 is SVM based, and System 4 is the combination of Sytems 2 and 3.

V. Conclusion

We studied approaches for detection of mispronunciations, based on the explicit acoustic modeling of correct and mispronounced training examples.

We proposed and analyzed two new mispronunciation detection algorithms. The first is based on computing the GMM likelihood ratio of adapted models to the correct and mispronounced training examples for each phone class. The second approach is based on discriminative modeling using as features supervectors derived from the parameters of adapted GMMs to phone segments, and SVM classifiers trained on examples of correct and mispronounced phones. Both methods proved to be superior to our previous mispronunciation detection system based on the LLR of independently trained GMMs.

The first approach produced a relative reduction of the weighted average EER of 3.5%, while the second approach produced a 4.7% relative reduction of the weighted average EER. Furthermore, a third system based on the combination of the scores provided by the first two systems produced an almost additive error reduction, resulting in an 8% relative reduction for the average EER.

References

[1] Y. Kim, H. Franco, and L. Neumeyer, “Automatic pronunciation scoring of specific phone segments for language instruction,” Proc. EUROSPEECH 97, pp. 649–652, Rhodes, 1997.

[2] J. Bernstein, M. Cohen, H. Murveit, D. Rtischev, and M. Weintraub, “Automatic evaluation and training in English pronunciation,” Proc. ICSLP 90, pp. 1185–1188. Kobe, Japan, 1990.

[3] L. Neumeyer, H. Franco, M. Weintraub, and P. Price, “Automatic text-independent pronunciation scoring of foreign language student speech,” Proc. ICSLP 96, Philadelphia, Pennsylvania, pp. 1457–1460, 1996.

[4] H. Franco, L. Neumeyer, Y. Kim, and O. Ronen, “Automatic pronunciation scoring for language instruction,” Proc. ICASSP 97, pp. 1471–1474, Munich, 1997.

[5] K. Precoda, C. Halverson, and H. Franco, “Effect of speech recognition-based pronunciation feedback on second language pronunciation ability,” Proc. InSTILL2000: Integrating Speech Technology in Learning, pp. 102-105. University of Albertay, Dundee, Scotland. 2000.

[6] M. Eskenazi, “Detection of foreign speakers’ pronunciation errors for second language training – preliminary results,” Proc. ICSLP 96, pp. 1465–1468. 1996.

[7] S. Witt and S. Young, ”Language learning based on non-native speech recognition,” Proc. EUROSPEECH 97, pp. 633–636, Rhodes, 1997.

[8] O. Ronen, L. Neumeyer, and H. Franco, “Automatic detection of mispronunciation for language instruction,” Proc. EUROSPEECH97, pp. 645-648, Rhodes, 1997.

[9] H. Franco, L. Neumeyer, M. Ramos, and H. Bratt, “Automatic detection of phone-level mispronunciations for language learning,” Proc. Eurospeech 99, 2, pp. 851-854, Budapest, Hungary, 1999.

[10] K. Truong, A. Neri, C. Cucchiarini, and H. Strik, “Automatic pronunciation error detection: An acoustic-phonetic approach,” Proc. InSTIL/ICALL Symposium, 17-19 June, Venice, Italy, pp. 135-138. 2004.

[11] H. Strik, K. Truong, F. de Wet, and C. Cucchiarini, “Comparing classifiers for pronunciation error detection,” Proc. Interspeech-2007, Antwerp, Belgium, pp. 1837-1840. 2007.

[12] D.A. Reynolds, T. F. Quatieri, and R.B. Dunn, “Speaker verification using adapted Gaussian mixture models,” Digital Signal Processing, 10, pp. 19-41. 2000.

[13] W. Campbell, D. Sturim, D. Reynolds, and A. Solomonoff, “SVM based speaker verification using a GMM supervector kernel and NAP variability compensation,” ICASSP 2006, Toulouse, France. May 15-19, 2006.

[14] H. Bratt, L. Neumeyer, E. Shriberg, and H. Franco, “Collection and detailed transcription of a speech database for development of language learning technologies,” Proc. ICSLP 98, pp. 1539-1542. Sydney, Australia. 1998.

[15] H. Franco, L. Neumeyer, Y. Kim, and O. Ronen, “Automatic pronunciation scoring for language instruction,” Proc. ICASSP 97, pp. 1471-1474, Munich. 1997.

[16] J. L. Gauvain and C.H. Lee, “Maximum a posteriori estimation for multivariable Gaussian mixture observatioins of Markov models,” IEEE Trans. on Speech and Audio Processing, 2(2), pp. 291-298, 1994.

[17] T. Joachims, “Making large-scale SVM learning practical,” Advances in Kernel Methods - Support Vector Learning, B. Schölkopf, C. Burges, and A. Smola (ed.), MIT Press, 1999.

[18] S. Siegel and N. John Castellan, Jr., Nonparametric Statistics for the Behavioral Sciences, Second Edition. New York: McGraw-Hill, 1988.

[19] H. Franco, V. Abrash, K. Precoda, H. Bratt, R. Rao, and J. Butzberger, “The SRI EduSpeak(TM) system: Recognition and pronunciation scoring for language learning,” Proc. InSTIL 2000, Dundee, Scotland, 2000.

[20] V. Digalakis, P. Monaco, P., and H. Murveit, Genones: Generalised mixture tying in continuous hidden Markov model-based speech recognizers, IEEE Trans. Speech and Audio Processing, vol. 4/4, pp. 281-289, 1996.

On the Benefit of Using Auditory Modeling for Diagnostic Evaluation of Pronunciations

Christos Koniaris, Olov Engwall and Giampiero Salvi
Centre for Speech Technology, School of Computer Science & Communication
KTH - Royal Institute of Technology, Stockholm, Sweden
{koniaris, engwall, giampi}@kth.se

***Abstract*—In this paper we demonstrate that a psychoacoustic model-based distance measure performs better than a speech signal distance measure in assessing the pronunciation of individual foreign speakers. The experiments show that the perceptual-based method performs not only quantitatively better than a speech spectrum-based method, but also qualitatively better, hence showing that auditory information is beneficial in the task of pronunciation error detection. We first present the general approach of the method, which is using the dissimilarity between the native perceptual domain and the non-native speech power spectrum domain. The problematic phonemes for a given non-native speaker are determined by the degree of disparity between the dissimilarity measure for the non-native and a group of native speakers. The two methods compared here are applied to different groups of non-native speakers of various language backgrounds and validated against a theoretical linguistic study.**

I. Introduction

Second language (L2) training involves acquiring the pronunciation of the target phonemes. In order to reach a correct L2 pronunciation, the learner must firstly perceive the auditory distinctiveness of the L2 phonemes, which may be problematic, especially when the foreign sounds are of a different phonological origin [1], [2]. Evidence of this is found in the learner's production through either large variations between attempts to produce the same phoneme (i.e., low precision) or consistent replacement of the target phoneme by another (i.e., low accuracy), often the most similar one in the native language (L1). Our objective is to make an automatic diagnostic evaluation of the phonemes that require additional practicing. For this, we propose a language independent, auditory model-based method that automatically identifies phonemes that are repeatedly mispronounced by individual non-native speakers in pre-collected recordings.

The novelty of our approach, previously presented in [3], lies in the utilization of auditory periphery as a tool to detect potential deviation in the foreign speakers pronunciation. Commonly, pronunciation error detection has been formulated as a classification problem. In [4] for example, the goodness of pronunciation (GOP) algorithm was presented to calculate the likelihood ratio of a phoneme realization to its canonical pronunciation. In [5], four different classifiers were examined to account for mispronunciation detection: a GOP-based, one combining cepstral coefficients and linear discriminant analysis, and two acoustic-phonetic classifiers. In [6], the problem was formulated within a support vector machine framework, with pronunciation space models to improve performance. Alternatively, articulatory information can been used to improve automatic detection of typical phoneme-level errors made by non-native speakers [7].

Unquestionably, the upper level processes of the primary auditory cortex inside the brain by which humans develop the ability to harmonize their hearing (and thereby, their production) system to the sounds of their L1 [8], [9] play an important part when explaining pronunciation difficulties in an L2, but this is out of the scope of this paper. Here, we instead concentrate on the native speakers' perception of non-native speech on the auditory level. Inconsistent variations or even deletion or replacement of the L2 phonemes may lead to difficulties for native listeners [10], [11].

In this paper, we evaluate the method we proposed in [3], for automatic diagnostic evaluation of the problematic phonemes for the L2 speaker, by comparing it with a different dissimilarity measure that does not incorporate auditory information. We first summarize the perceptual-based method as such and its application to pronunciation analysis. The central principle of the method is built upon measuring, for each phoneme, the similarity of the Euclidean geometry of the auditory representation for a group of native speakers and the speech signal's power spectrum for, on the one hand the native speaker group, and on the other individual non-native speakers. By comparing the measures for the non-native speaker and the native speakers, it is found, quantitatively, the phonemes that are mispronounced by the L2 speaker. The above is compared to a measure that only considers the Euclidean geometric similarities between the native and the non-native power spectrums. The two measures are evaluated with respect to a linguistic study [12].

II. Auditory processing of L1 speech signals

We form mathematically the task of pronunciation error detection with an objective to set up an automatic diagnostic evaluation scheme. We anticipate the existence of the *perceptual subspace*, i.e., the area with high sensitivity to speech signal changes [14], [15]. We establish a measure of similarity for a non-native speech vector based on perturbation analysis and distortion criteria derived from a psychoacoustic model that is trained by native speech. We investigate which aspects of the non-native perturbation signal are lying inside the native perceptual subspace, and evaluate the range of discrepancy

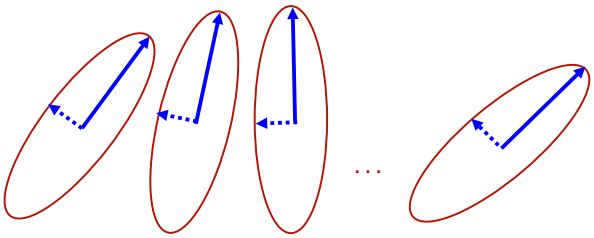


Fig. 1. Orientation of the auditory distortion measure domain (red ellipses) and the directions of the perturbation vectors, in parallel (highest sensitivity-solid arrows) and orthogonal (lowest sensitivity-dotted arrows) to it.

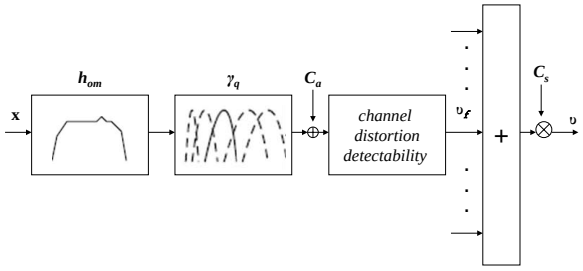


Fig. 2. Block diagram of a channel of the van de Par auditory model [13].

as compared to the native speech data. Fig. 1 illustrates the relation between the distortion measure and the perturbations. The ellipses denote the psychoacoustic distortion measure sensitivity orientations – as indicated by the transformation applied to the signal – over each perturbation. The arrows show the directions of the perturbation vectors. Distortions that follow the orientation of the psychoacoustic distortion measures (parallel vectors) are the most perceptually relevant to the auditory periphery, which implies that every small change in the speech signal is detected and identified by the hearing system. The orthogonal arrows denote the direction with the minimum response. A small alteration (error) in the signal associated to these specific perturbation vectors are not captured by the auditory system.

A. Exploiting auditory knowledge

Models of the auditory periphery are used in several cases to examine the perceptual processing of a sound signal or to explain the way it operates. We consider a spectral psychoacoustic model [13], known as the van de Par model, that uses a series of auditory filters for computing the total distortion in speech, a characteristic that is derived from the spectral integration property of the human auditory system. It is divided into three major parts as shown in Fig. 2, namely the outer, the middle and the inner ear, which simulate the corresponding components of the peripheral system. The first two are represented by a band-pass filter followed by a gammatone filterbank that models the basilar membrane of the inner ear. The center frequencies of the gammatone filterbank are spaced linearly on a equivalent rectangular bandwidth scale. At the end, the total distortion measure is calculated as a summation of the distortion detectability provided by each auditory filter f multiplied by the effective duration L_e of the input signals.

In [14], [15] a distortion measure in the sound perception domain was defined as $\upsilon : \mathbb{R}^N \times \mathbb{R}^N \rightarrow \mathbb{R}^+$. Consider $\mathbf{y} : \mathbb{R}^N \rightarrow \mathbb{R}^M$ to be a mapping of the power spectrum to the M-dimensional perceptual domain. Then, allow $\mathbf{y}(\mathbf{x}_i)$ and $\mathbf{y}(\hat{\mathbf{x}}_{i,j})$ to be the auditory model output signals, where $\mathbf{x}_i$ is the power spectrum signal of frame $i \in \mathbb{Z}$ and $\hat{\mathbf{x}}_{i,j}$ its j'th perturbation, respectively. Thus $\upsilon(\mathbf{x}_i, \hat{\mathbf{x}}_{i,j}) = \| \mathbf{y}(\mathbf{x}_i) - \mathbf{y}(\hat{\mathbf{x}}_{i,j}) \|^2$. The van de Par model provides a measure to detect distortion in the perceptual domain without explicitly computing the auditory model output signals $\mathbf{y}(\mathbf{x}_i)$ and $\mathbf{y}(\hat{\mathbf{x}}_{i,j})$. Thus, combining the perturbation theory and the sensitivity matrix [16], we make the approximation $\upsilon(\mathbf{x}_i, \hat{\mathbf{x}}_{i,j}) \approx [\hat{\mathbf{x}}_{i,j} - \mathbf{x}_i]^T \mathbf{D}_\upsilon(\mathbf{x}_i)[\hat{\mathbf{x}}_{i,j} - \mathbf{x}_i]$, where $\mathbf{D}_{\upsilon,\mu\nu}(\mathbf{x}_i) = \frac{\partial^2 \upsilon(\mathbf{x}_i, \hat{\mathbf{x}}_{i,j})}{\partial \hat{x}_\mu \partial \hat{x}_\nu}\Big|_{\hat{\mathbf{x}}_{i,j}=\mathbf{x}_i}$ is the sensitivity matrix. This matrix can be calculated using the van de Par model. The word 'sensitivity' refers to the fact that each element of this matrix represents the sensitivity of the distortion $\upsilon(\mathbf{x}_i, \hat{\mathbf{x}}_{i,j})$ to a particular $[\hat{\mathbf{x}}_{i,j} - \mathbf{x}_i]$.

III. L2 PRONUNCIATION ANALYSIS

Fig. 3 shows a block diagram of the two considered methods. An HMM-based aligner [17] generates a phone-level transcription from the speech signal and the text file, that separates the native speech stimuli into phoneme categories. For each considered phoneme class, the native signal is transformed into the auditory and the power spectrum representations. The spatial dissimilarity

$$\mathcal{A}_n = \frac{1}{\mathcal{I}} \sum_{i \in \mathcal{I}} \frac{1}{\mathcal{J}_i} \sum_{j \in \mathcal{J}_i} \left[\upsilon_n(\mathbf{x}_{n_i}, \hat{\mathbf{x}}_{n_{i,j}}) - \phi_n(\mathbf{x}_{n_i}, \hat{\mathbf{x}}_{n_{i,j}}) \right]^2, \quad (1)$$

between these two domains is measured to investigate, quantitatively, to what extent the native speech signal representations $\mathbf{x}_n$ include all the information relayed by the human auditory periphery. In the above Eq. (1), $\phi(\cdot)$ is the power spectrum distortion measure defined as $\phi : \mathbb{R}^N \times \mathbb{R}^N \rightarrow \mathbb{R}^+$, where $\mathbb{R}^+$ are the non-negative real numbers, which is a Euclidean norm-based measure calculated by $\phi(\mathbf{x}_i, \hat{\mathbf{x}}_{i,j}) = \| \mathbf{x}_i - \hat{\mathbf{x}}_{i,j} \|^2$. By allowing small distortions, the dissimilarity of the Euclidean geometry is calculated between the magnitude spectrum of the speech and the auditory periphery output to establish what we call the *native perception*. The non-native stimuli $\mathbf{x}_\ell$ is then considered, but in this case only the power spectrum transformation is calculated. The dissimilarity measure

$$\mathcal{A}_\ell = \frac{1}{\mathcal{I}} \sum_{i \in \mathcal{I}} \frac{1}{\mathcal{J}_i} \sum_{j \in \mathcal{J}_i} \left[\upsilon_n(\mathbf{x}_{n_i}, \hat{\mathbf{x}}_{n_{i,j}}) - \phi_\ell(\mathbf{x}_{\ell_i}, \hat{\mathbf{x}}_{\ell_{i,j}}) \right]^2, \quad (2)$$

is then computed between the native perceptual distortion and the non-native power spectrum distortion, and the two measures $\mathcal{A}_n$ and $\mathcal{A}_\ell$ are compared to identify the problematic phonemes.

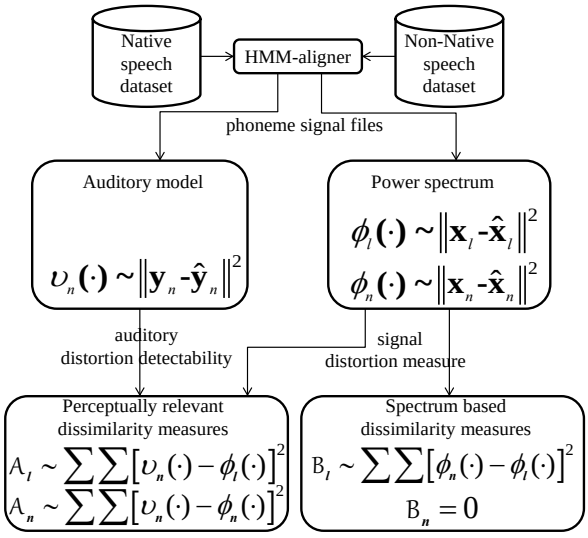


Fig. 3. Block diagram of the two methods. The L^2 norm of the phone distortion is evaluated in two different domains, the auditory and the power spectrum in the first method, and only in the power spectrum in the second method. The objective is to measure the Euclidean geometry dissimilarity in the considered domains.

The second method finds directly the geometric dissimilarity

$$\mathcal{B}_\ell = \frac{1}{\mathcal{I}} \sum_{i \in \mathcal{I}} \frac{1}{\mathcal{J}_i} \sum_{j \in \mathcal{J}_i} \left[\phi_n(\mathbf{x}_{n_i}, \hat{\mathbf{x}}_{n_{i,j}}) - \phi_\ell(\mathbf{x}_{\ell_i}, \hat{\mathbf{x}}_{\ell_{i,j}}) \right]^2 \quad (3)$$

between the native and the non-native power spectrum distortion. In this case, $\mathcal{B}_n$ is zero.

Eqs. (2)-(3) compute the dissimilarity measures by considering speech from different sources, native and non-native. It is thus necessary to find a relationship between the two speech signals. In the following section we describe a way to achieve this.

A. Local linearization of non-native speech

Let $\mathbf{p}$ be a phone that represents a target phoneme category produced by a speaker. Let then $\mathbf{x}(\mathbf{p})$ denote the speech power spectrum as a function of $\mathbf{p}$. Assuming the mapping $\mathbf{x}$ to be analytic, the Taylor series can be used to make a local approximation around $\mathbf{p}$:

$$\mathbf{x}(\hat{\mathbf{p}}) \approx \mathbf{x}(\mathbf{p}) + \mathbf{J}_\mathbf{x}[\hat{\mathbf{p}} - \mathbf{p}], \quad (4)$$

where $\mathbf{J}_\mathbf{x} = \frac{\partial \mathbf{x}(\mathbf{p})}{\partial \hat{\mathbf{p}}}\Big|_{\hat{\mathbf{p}}=\mathbf{p}}$. Eq. (4) can be applied to both native speech $\mathbf{x}_n$ and non-native speech $\mathbf{x}_\ell$ of a language background ℓ. In both cases, the distortion $[\hat{\mathbf{p}}-\mathbf{p}]$ remains common, allowing us to find a linearized relation between these two and compute the speech power spectrum distortion in a non-native subspace into the native speech power spectrum domain. In this case we get

$$\mathbf{x}_\ell(\hat{\mathbf{p}}) \approx \mathbf{x}_\ell(\mathbf{p}) + \mathbf{W}_\ell \left[\mathbf{x}_n(\hat{\mathbf{p}}) - \mathbf{x}_n(\mathbf{p}) \right], \quad (5)$$

where $\mathbf{W}_\ell = \mathbf{J}_{\mathbf{x}\ell} \left[\mathbf{J}_{\mathbf{x}n} \right]^{-1}$. Then, the power spectrum distortion measure for the non-native speech signal is calculated as

$$\phi_\ell(\mathbf{x}_{\ell_i}, \hat{\mathbf{x}}_{\ell_{i,j}}) \cong \phi_\ell(\mathbf{x}_{n_i}, \hat{\mathbf{x}}_{n_{i,j}}; \mathbf{x}_{\ell_i}, \hat{\mathbf{x}}_{\ell_{i,j}}) \approx \left[\mathbf{x}_{n_i} - \hat{\mathbf{x}}_{n_{i,j}} \right]^T \left[\mathbf{W}_\ell \right]^T \mathbf{W}_\ell \left[\mathbf{x}_{n_i} - \hat{\mathbf{x}}_{n_{i,j}} \right], \quad (6)$$

where $i \in \mathcal{I}$, $j \in \mathcal{J}_i$.

Eq. (6) implies the calculation of the $\mathbf{W}_\ell$ matrix on a frame basis. However, there are practical and theoretical reasons that prohibit such a consideration. Firstly, phone duration or silence mismatch between the native and non-native speech signal preclude this option. Furthermore, mathematically speaking, the matrices are non-invertible, and hence requiring an alternative approach. We therefore compute $\mathbf{W}_\ell$ by considering a common matrix for all frames i of a specific foreign language group of speakers ℓ. We assume both native and non-native speech signals to comply with a Gaussian distribution. Thus, Eq. (5) can be expressed as $\mathcal{N}(\mu_\ell,\ \mathbf{\Sigma}_\ell) \sim \mathcal{N}(\mathbf{W}_\ell \mu_n,\ \mathbf{W}_\ell \mathbf{\Sigma}_n \left[\mathbf{W}_\ell\right]^T)$, where μ_ℓ, μ_n are the mean vectors of the distortion in non-native and native speech signals for a phoneme class p, respectively and $\mathbf{\Sigma}_\ell$, $\mathbf{\Sigma}_n$ their covariance matrices. It can be shown [3] that the matrix $\mathbf{W}_\ell$ is given by

$$\mathbf{W}_\ell = \mathbf{V}_\ell \left[\mathbf{S}_\ell\right]^{\frac{1}{2}} \left[\mathbf{S}_n\right]^{-\frac{1}{2}} \left[\mathbf{V}_n\right]^T, \quad (7)$$

where $\mathbf{V}_\zeta, \mathbf{S}_\zeta$ are the matrices that decompose the corresponding covariance matrices $\mathbf{\Sigma}_\zeta = \mathbf{V}_\zeta \mathbf{S}_\zeta \left[\mathbf{V}_\zeta\right]^T$ of the above Gaussian distributions with $\zeta \in \{n, \ell\}$ for the native language group and the non-native language group, respectively.

B. The algorithm

The central idea of the method is to create a local geometry around each speech sound by allowing small perturbations. In practice, this is done by adding 30 dB SNR i.i.d. Gaussian noise to each $\mathbf{x}_i$ and generate a set of 100 vectors $\hat{\mathbf{x}}_{i,j}$ for the native speech data n as well as for non-native speech data of all language backgrounds ℓ. Considering the native speech, for each speech segment i, the sensitivity matrix $\mathbf{D}_{\upsilon_n}(\mathbf{x}_{n_i})$ is calculated using the van de Par model on a frame basis with the objective to compute the perceptual distortion measure for the native speech signal. Eq. (7) is used to compute the matrix $\mathbf{W}_\ell$ over all frames of each phoneme and language group. Next, the dissimilarity measure $\mathcal{A}_\ell$ is calculated using the native perceptual distortion measure $\upsilon_n(\mathbf{x}_{n_i}, \hat{\mathbf{x}}_{n_{i,j}}) \approx [\hat{\mathbf{x}}_{n_{i,j}} - \mathbf{x}_{n_i}]^T \mathbf{D}_{\upsilon_n}(\mathbf{x}_{n_i})[\hat{\mathbf{x}}_{n_{i,j}} - \mathbf{x}_{n_i}]$, and the non-native speech frequency distortion measure $\phi_\ell(\mathbf{x}_{n_i}, \hat{\mathbf{x}}_{n_{i,j}}; \mathbf{x}_{\ell_i}, \hat{\mathbf{x}}_{\ell_{i,j}})$ given by Eq. (6). Then, the corresponding dissimilarity measure for the native speakers $\mathcal{A}_n$ is calculated using again $\upsilon_n(\mathbf{x}_{n_i}, \hat{\mathbf{x}}_{n_{i,j}})$ and the native speech frequency distortion measure $\phi_n(\mathbf{x}_{n_i}, \hat{\mathbf{x}}_{n_{i,j}}) = \| \mathbf{x}_{n_i} - \hat{\mathbf{x}}_{n_{i,j}} \|^2$. The perceptual based method ends by considering for each phoneme class, the native-perceptual assessment degree[1]

[1]It is a normalized ratio that shows the degree of the dissimilarity between the native perceptual outcome and the non-native power spectrum as compared to the native-only case.

TABLE I
DISTRIBUTION OF TOTAL NUMBER OF MALE AND FEMALE SPEAKERS AND NUMBER OF UTTERANCES FOR EACH LANGUAGE BACKGROUND.

L1 background	total	male/female	utterances	L1 background	total	male/female	utterances	L1 background	total	male/female	utterances
Eng.(US)	2	1/1	318	*Russian*	4	1/3	583	*Arabic*	1	0/1	164
German	2	2/0	249	*Greek*	3	3/0	393	*Chinese*	5	2/3	832
French	3	3/0	347	*Spanish*	5	4/1	882	*Persian*	6	3/3	987
Polish	2	0/2	317	*Turkish*	4	4/0	604	*Swedish*	11	9/2	888

nPAD Θ_ℓ that is computed for every L1 background as

$$\Theta_\ell = \frac{\mathcal{A}_\ell}{\mathcal{A}_n}, \tag{8}$$

considering a mispronunciation to have occurred when $\Theta_\ell > 1$. On the other hand, for the spectral based method the power spectrum distortion measure $\mathcal{B}_\ell$, Eq. (3), is evaluated using the native speech frequency distortion measure $\phi_n(\mathbf{x}_{n_i}, \hat{\mathbf{x}}_{n_{i,j}})$ and the non-native speech frequency distortion measure $\phi_\ell(\mathbf{x}_{n_i}, \hat{\mathbf{x}}_{n_{i,j}}; \mathbf{x}_{\ell_i}, \hat{\mathbf{x}}_{\ell_{i,j}})$ given by Eq. (6).

IV. EXPERIMENTAL EVALUATION

In this section, we describe the data used for our experiments, and present the experimental setup and our evaluation outcome.

A. Speech corpus

The speech database was collected from L2 learners of Swedish using a computer-assisted language learning program Ville [18], in which an embodied conversational agent plays the role of a language teacher. The corpus, sampled at 16 kHz, consists of data from 37 (23 male and 14 female) speakers of different language backgrounds (see Table I), who took the test twice within one month's time, before and after practising at home. Each session lasted 30 minutes composed of exercises in which the participants repeated single words and sentences of varying complexity after Ville. In order to provide a native speaker reference, 11 (9 males and 2 females) Swedish speakers were also recorded once each.

The data were analyzed and stripped from any non-linguistic content, e.g., coughs, long pauses and hesitation phenomena, such as repetitions and fillers ("um", "uh", "eh" etc). The text file that accompanies each sound file was adjusted to the actual content of the included speech utterance when deletions or insertions happened to occur. Using the clean audio and text files, the data was divided into phoneme categories, using a HMM-based aligner [17]. The material contained all Swedish phonemes, but the two short and more open pre-r allophones /æ/, /œ/ and the retroflexes /ɳ/, /ɖ/, and /ʂ/ were not considered because the number of occurrences in the database was not sufficiently large. Finally, the speech signal was pre-emphasized and the output was windowed by a Hamming window of 25 ms with an overlap of 10 ms. A discrete Fourier transform of 512 points was applied to the windowed frame to compute the signal's power spectrum.

B. Results

We present the results of our experiments and argue for the use of the nPAD Θ_ℓ against the spectral dissimilarity measure $\mathcal{B}_\ell$. In addition, we compare our findings with Bannert's linguistic study of problematic Swedish phonemes of different L1 groups [12]. Naturally, the two approaches presented here differ from [12] insofar as we only consider the acoustic signal of the uttered phonemes to evaluate the foreign accent, without intending to perform a comprehensive linguistic study. That being so, any grammatical or syntactical errors, as well as errors due to context, are excluded.

Table II lists the phonemes found to be difficult for the different groups of non-native speakers according to the two considered methods. For each L2 speaker group, the first line shows, in order, the most deviating vowels to the left, and consonants to the right, according to the nPAD Θ_ℓ. Correspondingly, the second line shows the evaluation of the spectral dissimilarity measure $\mathcal{B}_\ell$. The results of our perceptual-based method are, in general, in better agreement with previous linguistic observations [12]. Divergences from the theoretical findings are reported in parentheses. It can be seen that the majority of the parentheses are located on the right side meaning that most of the disagreements to the theoretical results are for lower Θ_ℓ. In general, this also holds for $\mathcal{B}_\ell$ but it is worth mentioning that for three language groups the first vowels are misdetections according to [12]. Comparing the algorithms, we see that, generally speaking, for as many as eight language groups, i.e., *English US, German, French, Polish, Greek, Turkish, Arabic,* and *Persian* the nPAD method gets quantitatively better results. For three language groups, namely *Russian, Spanish,* and *Chinese*, the results of the perceptual-based method become worse in comparison to the spectral dissimilarity measure. The perceptual-based method not only performs better in terms of a lower number of disagreements of detected problematic phonemes compared to the linguistic study, but also in terms of detection of seriously mispronounced phonemes (i.e., those that are totally mispronounced, as described in [12] and also listed in [19]). In short, the nPAD method has one mismatch less for the German and Arabic speakers, two less for French, Greek and Turkish speakers and three less for the English speakers. In addition, concentrating mainly on the seriously problematic phonemes, nPAD captures one more seriously problematic phoneme for German, Polish, Greek, Arabic and Persian speakers, two more for French and three for Turkish speakers.

TABLE II
PROBLEMATIC PHONEMES PER LANGUAGE BACKGROUND. THE PHONEMES ARE SHOWN IN DECREASING ORDER, STARTING FROM THE ONE WITH THE HIGHEST Θ_ℓ OR $\mathcal{B}_\ell$ DEPENDING ON THE EVALUATION METHOD USED. PHONEMES THAT DIFFER FROM THE LINGUISTIC STUDY FINDINGS ARE LISTED IN PARENTHESES.

L1 background	evaluation	vowels	consonants
English (US)	Θ_ℓ	æː, ɛ, yː, uː, ʊ, œː, ɛː, ø, ɵ, øː, (iː), ɑː, (ə), eː, e, ɔ, a, ʉː	ʃj, ŋ, (v), m, n, (b), r, (d), l, k, ʂ, t
	$\mathcal{B}_\ell$	ø, ɛ, ɑː, æː, a, œː, ɛː, e, øː, eː, ʏ, (i), ɔ, (iː), (ə), ʉː, yː, ɵ	ʂ, ç, (d), s, (b), (j), ʃj, (g), (h), r, t, ŋ
German	Θ_ℓ	æː, (ɛ), yː, uː, (ʊ), ɛː, (ø), œː, øː, iː, ə, ɑː	ʃj, ŋ, v, n, (m), b, r, d, (l), k, ʂ, t, p, (h), f, ç, s
	$\mathcal{B}_\ell$	(ø), æː, ɑː, (ɛ), œː, (a), øː, ɛː, (e), ɵ, ʏ, (i)	ç, ʂ, s, t, k, g, (l), (h), ʃj, r, d, b, f, j, v, n, ŋ
French	Θ_ℓ	æː, ɛ, yː, uː, œː, (ʊ), ɛː, (ø), ɵ, øː, iː, ə, ɑː, eː, e, ɔ, (a)	ŋ, ʃj, (v), m, n, b, r, (l), d, ʂ, k, t, p, h, ç, g
	$\mathcal{B}_\ell$	(ø), ɛ, ɑː, æː, (a), œː, ɛː, e, øː, eː, ʏ, (i), ɔ, iː, ə, ʉː, yː	ç, ʂ, s, k, t, (l), g, h, (j), r, b, d, ʃj, (f), p, (v)
Polish	Θ_ℓ	æː, (ɛ), yː, uː, œː, (ʊ), ɛː, ø, ɵ, øː, iː, ɑː, ə, eː, (e), (ɔ)	ʃj, ŋ, v, (m), n, b, (r), d, (l), k, ʂ, t, p, s, (f)
	$\mathcal{B}_\ell$	ø, (ɛ), ɑː, æː, a, œː, ɛː, (e), øː, eː, ʏ, (i), (ɔ), iː, ə, ʉː	ç, ʂ, s, t, k, g, (l), ʃj, h, (j), (r), b, d, (f), v
Russian	Θ_ℓ	ɛ, yː, ɛː, ø, øː, iː, ɑː, ɵ, eː, e, ʉː, a, (ɔ), ʏ, (ə), (i), œː	v, ŋ, (m), (n), (r), d, (l), h, b, k, t, g, (s), f, j
	$\mathcal{B}_\ell$	ø, œː, ɛː, øː, ʏ, ɑː, a, e, ɛ, æː, eː, ʉː, uː, yː, oː, iː, ɵ	ç, ʃj, ʂ, g, (s), k, b, ŋ, h, f, t, (l), (r), (m), j
Greek	Θ_ℓ	æː, (ɛ), yː, uː, œː, (ʊ), ɛː, ø, ɵ, øː, iː, ɑː, ə, eː, e, (ɔ)	ʃj, ŋ, (v), m, n, b, (r), d, l, ʂ, k, t, p, ç, (f), s
	$\mathcal{B}_\ell$	ø, (ɛ), ɑː, æː, (a), œː, ɛː, e, øː, eː, ʏ, (i), (ɔ), iː, ə, ʉː	ç, ʂ, s, t, k, g, l, ʃj, (j), (r), b, d, h, (f), (v), n
Spanish	Θ_ℓ	uː, æː, ɛ, ø, iː, ɵ, eː, e, øː, (a), ʉː, ɑː, (i), ə, ʏ, (ɔ), oː	ŋ, v, (r), n, (l), b, t, ʂ, (f), k, g, d, j, p, ç, s, h
	$\mathcal{B}_\ell$	uː, æː, yː, ø, (a), œː, e, (ʊ), ɛ, oː, ɛː, eː, (i), iː, ʏ, ʉː, øː	ʃj, ç, s, h, ʂ, (l), b, ʈ, (r), (f), k, d, p, t, j, g, ŋ
Turkish	Θ_ℓ	æː, (ɛ), yː, (ɛː), (ø), uː, ɵ, ʊ, øː, iː, ɑː, eː, (ə)	ŋ, v, (m), n, b, r, l, d, k, (ʂ), t, p, f, h, g, ʈ, j, s
	$\mathcal{B}_\ell$	(ø), (ɛ), æː, (a), ɑː, œː, (ɛː), øː, e, eː, (ʏ), (i), (ɔ)	ç, (ʂ), t, ʃj, k, s, l, g, h, r, j, b, d, f, v, n, p, ŋ
Arabic	Θ_ℓ	æː, ɛ, yː, uː, œː, (ʊ), ɛː, ø, ɵ, øː, iː, ɑː, ə, eː, e, (ɔ), a, ʉː	ʃj, ŋ, v, (m), (n), (b), r, d, (l), k, ʂ, t, p
	$\mathcal{B}_\ell$	ø, ɛ, ɑː, æː, a, œː, ɛː, e, øː, eː, ʏ, (i), (ɔ), iː, ə, ʉː, yː, oː	ç, ʂ, s, t, k, (g), (l), ʃj, (h), (j), r, (b), d
Chinese	Θ_ℓ	ɵ, æː, ɛ, yː, uː, ɛː, ø, øː, iː, ɑː, eː, e, ɔ, (ə), a, ʉː, (i), oː	ʃj, ŋ, v, m, n, b, r, l, d, k, t, f, g, ʈ, p, j, (h), (s)
	$\mathcal{B}_\ell$	ɵ, ʏ, ø, ɑː, æː, ɛ, œː, a, e, ɛː, øː, (i), eː, ɔ, iː, ʉː, oː, yː	ç, ʂ, (s), d, b, t, k, ʃj, g, (h), j, l, r, t, ʈ, f, v, n
Persian	Θ_ℓ	ɵ, æː, yː, (ɛ), ø, øː, ʏ, (i), ʉː, eː, a, (e), (ɑː), oː, (ɔ)	b, d, (ʃj), v, (f), g, (h), t, s, (j), ç, p, ʈ, k, l, r
	$\mathcal{B}_\ell$	ɵ, æː, yː, oː, ə, a, (ʊ), iː, (e), (ɔ), ø, (ɑː), (ɛ), eː, uː	ç, ʂ, b, (ʃj), s, g, (h), k, t, r, d, (j), (f), l, v, ʈ

The spectral dissimilarity measure has one mismatch less for Chinese speakers and four for Russian speakers. Finally, it captures one more seriously problematic phoneme for English, Spanish and Chinese speakers and two more for Russian speakers. To summarize, the nPAD method has in total 69 mismatches (34 for vowels and 35 for consonants) and the spectral measure 75 (37 for vowels and 38 for consonants) out of 350 mispronunciations listed in [12]. Finally, the nPAD method has 49 missed seriously problematic phonemes (27 for vowels and 22 for consonants) and the spectral measure 54 (34 for vowels and 20 for consonants) out of 245 in total.

Even though the performance of nPAD is better than that of the spectral evaluation, it appears to have some disagreements with the theoretical study concerning the problematic phonemes. This can be explained by the nature and the context of the data since the recordings were made with the subjects repeating, in two sessions, text after a natively speaking virtual language tutor. Hence, it is likely that the speakers have avoided some otherwise occurring mispronunciations, that usually accompanying spontaneous speech.

C. Discussion

Looking at the Table II we can recognize some important weaknesses of the measure $\mathcal{B}_\ell$ in identifying major mispronunciations for some of the most problematic phonemes. In most cases for example, it does not recognize the Swedish long rounded vowel /uː/ as problematic, except for the Spanish group. The linguistic findings have shown that all foreign groups mispronounce the Swedish /uː/ because they produce it either as a short vowel or with inadequate lip rounding. The nPAD measure on the other hand not only is able to capture this vowel, but additionally to rank it high on the problematic vowels list. In most cases, except for Chinese and Persian speakers, the spectral distortion measure $\mathcal{B}_\ell$ further fails in detecting the vowel /ɵ/ which is one of the seriously problematic Swedish vowels for non-native speakers, often confused by /ʊ/ [12]. Additionally, the measure $\mathcal{B}_\ell$ misses the vowel /yː/, which is produced with protruded instead of compressed lips, and according to [12], is mainly confused with the short unrounded /i/. On the contrary, the nPAD measure Θ_ℓ succeeds to detect both the aforementioned vowels and indeed to classify them among the most problematic vowels.

There are several consonants that appear to be problematic for many foreign speakers, four of which are in particular difficult for almost all of the language groups. The velar nasal /ŋ/ is one of the consonants that most speakers are inclined to mispronounce. Bannert has noticed that it is often replaced by /ŋg/. Table II shows that the nPAD method can better detect this error than the spectral distortion measure. Additionally, the Swedish consonant /v/, which is very often mispronounced by

non-native speakers either as /f/, /b/ or /w/, is also detected by the nPAD measure. The Swedish fricatives /ʃ/ and /ç/ are probably the most difficult Swedish consonants for non-native speakers due to their uniqueness and their large variety depending on the neighboring sounds. The nPAD approach is more capable in capturing the problems related to the /ʃ/, while the spectral distortion measure is more sensible in detecting errors only related to /ç/.

This significant superiority of the nPAD measure to detect problematic phonemes has, in our opinion, its roots in the information that the auditory model provides through the sensitivity matrix. It seems that the small distances in the power spectrum between the native and the non-native speech signals become clearer in the auditory domain where only the perceptually relevant elements of the two spectrums are considered. This enriches the nPAD method's potential to identify the meaningful details that reveal the pronunciation divergences between native and non-native speakers. In addition, we note that when computing the total value of $\mathcal{B}_\ell$ for all of the problematic consonants for each language, the part that corresponds to the value of the first one is very high. In other words, the method has limited capability in detecting the problematic consonants in general and can mainly focus on the detection of the most mispronounced consonant.

The use of the auditory knowledge is evidently improving the performance of our automatic pronunciation error detection system. Still, we believe that the method could be further developed by including a vital component of the speech signal, namely its dynamics. For this, two changes have to take effect; the auditory model to account for both the spectral and temporal aspects of the speech signal and the speech representations to include dynamic information (velocity and acceleration feature vectors). For some phonemes, e.g., the stop consonants, the onset detection is important in identifying pronunciation divergence from native speakers. As this onset does not always correspond to the perceptual onset, a phenomenon that arises from the temporal integration in the human auditory system [20], the nPAD method should be expanded in this direction. Finally, we believe that a proper evaluation of the method should include the judgement of native listeners on the same data, and we therefore plan to include such an evaluation in our future work.

V. Conclusions

In this paper, we investigated the benefit of using an auditory-based method to perform an automatic and language independent diagnostic assessment of phoneme pronunciation by non-native speakers. We examined the method by comparing it with a speech power spectrum dissimilarity measure that accounts only for the differences between the native and non-native speech signals, using a theoretical linguistic study as reference. The experiments show that the use of perceptual knowledge in identifying the common characteristics of the native speech signal, is beneficial for the task of L2 pronunciation error detection.

Acknowledgment

This work is supported by the Swedish Research Council project 80449001 Computer-Animated LAnguage TEAchers (CALATEA).

References

[1] J. E. Flege, "Second-language speech learning: theory, findings, and problems", *Strange, W. (Ed.), Speech Perception and Linguistic Experience: Theoretical and Methodological Issues in Cross-Language Speech Research. Timonium, MD: York Press Inc.*, pp. 233–272, 1995.

[2] S. G. Guion and J. E. Flege and R. Ahahane-Yamada and J. C. Pruitt, "An investigation of current models of second language speech perception: the case of Japanese adults' perception of English consonants", *J. Acoust. Soc. Amer.*, vol. 107, no. 5, pp. 2711–2724, May 2000.

[3] C. Koniaris and O. Engwall, "Phoneme level non-native pronunciation analysis by an auditory model-based native assessment scheme", in *Interspeech, Florence, Italy*, pp. 1157–1160, Aug. 2011.

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

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

[6] S. Wei, G. Hu, Y. Hu, and R.-H. Wang, "A new method for mispronunciation detection using support vector machine based on pronunciation space models", *Speech Communication*, vol. 51, no. 10, pp. 896–905, Oct. 2009.

[7] J. Tepperman and S. Narayanan, "Using articulatory representations to detect segmental errors in nonnative pronunciation", *IEEE Tr. Audio, Speech, Lang. Proc.*, vol. 16, no. 1, pp. 8–22, Jan. 2008.

[8] J. F. Werker and R. C. Tees, "Cross-language speech perception: Evidence for perceptual reorganization during the first year of life", *Infant Behavior and Development*, vol. 7, no. 1, pp. 49–63, Mar. 1984.

[9] P. K. Kuhl, "Early linguistic experience and phonetic perception: implications for theories of developmental speech perception", *J. Phonetics*, vol. 21, pp. 125–139, Jan.-Apr. 1993.

[10] M. J. Munro and T. M. Derwing, "Foreign accent, comprehensibility, and intelligibility in the speech of second language learners", *Language Learning*, vol. 45, no. 1, pp. 73–97, Mar. 1995.

[11] P. M. Schmid and G. H. Yeni-Komshian, "The effects of speaker accent and target predictability on perception of mispronunciations", *J. Speech, Lang., Hear. Res.*, vol. 42, pp. 56–64, Feb. 1999.

[12] R. Bannert, "Problems in learning Swedish pronunciation and in understanding foreign accent", *Folia Linguistica*, vol. 18, no. 1-2, pp. 193–222, Jan. 1984.

[13] S. van de Par, A. Kohlrausch, G. Charestan, and R. Heusdens, "A new psychoacoustical masking model for audio coding applications", in *IEEE Int. Conf. Acoust., Speech, Sig. Proc., Orlando, FL, USA*, vol. 2, pp. 1805–1808, May 2002.

[14] C. Koniaris, M. Kuropatwinksi, and W. B. Kleijn, "Auditory-model based robust feature selection for speech recognition", *J. Acoust. Soc. Amer.*, vol. 127, no. 2, pp. EL73–EL79, Feb. 2010.

[15] C. Koniaris, S. Chatterjee, and W. B. Kleijn, "Selecting static and dynamic features using an advanced auditory model for speech recognition", in *IEEE Int. Conf. Acoust., Speech, Sig. Proc., Dallas, TX, USA*, pp. 4342–4345, Mar. 2010.

[16] W. R. Gardner and B. D. Rao, "Theoretical analysis of the high-rate vector quantization of LPC parameters", *IEEE Tr. Speech, Audio Proc.*, vol. 3, no. 5, pp. 367–381, Sep. 1995.

[17] K. Sjölander, "An HMM-based system for automatic segmentation and alignment of speech", in *Fonetik*, pp. 93–96, Jun. 2003.

[18] P. Wik and A. Hjalmarsson, "Embodied conversational agents in computer assisted language learning", *Speech Communication*, vol. 51, no. 10, pp. 1024–1037, Oct. 2009.

[19] C. Koniaris and O. Engwall, "Perceptual differentiation modeling explains phoneme mispronunciation by non-native speakers", in *IEEE Int. Conf. Acoust., Speech, Sig. Proc., Prague, Czech Republic*, pp. 5704–5707, May 2011.

[20] D. A. Eddins and D. M. Green, "Temporal integration and temporal resolution. In Hearing by B. C. J. Moore", *Academic Press, San Diego, CA, USA. ISBN:0-12-505626-5*, pp. 207–242, Aug. 1995.

Automatic Classification of Pronunciation Errors Using Decision Trees and Speech Recognition Technology

Amalia Zahra, João P. Cabral, Mark Kane, Julie Carson-Berndsen
CNGL, School of Computer Science and Informatics, University College Dublin
Belfield, Dublin 4, Ireland
{amalia.zahra, mark.kane}@ucdconnect.ie, {joao.cabral, julie.berndsen}@ucd.ie

***Abstract*—Pronunciation error detection plays an important role in a pronunciation learning system. A method to categorise the pronunciation errors into three classes, obvious, minor, and no pronunciation error, is proposed in this paper. This work is concerned only with non-native speakers who would like to improve their English pronunciation. The method implemented here combines a decision tree and speech recognition technology, where no thresholds and rules are required. The experiment is carried out at the phone level. However the feedback to users is presented at the syllable level in order to ease their effort in learning English pronunciation. A subjective evaluation has been carried out at the syllable level. Based on the outcomes of the subjective evaluation performed by 12 participants (English native speakers), the false positive rate is 20.7% and the false negative rates are 17% and 33.7% for obvious and minor pronunciation error, respectively.**

I. Introduction

Computer-Aided Pronunciation Learning (CAPL) systems have been explored for more than three decades [1]. This is due to the increasing number of second language learners (L2 learners) who expect to be able to learn anytime and anywhere, without the presence of human teachers. There are two concerns regarding the usefulness of a pronunciation learning system: its ability to detect pronunciation errors and the appropriate feedback that can support the learning experience.

A number of studies in this area have been carried out. A speech-enabled CAPL for teaching Arabic pronunciations [2] to non-native speakers was developed by applying a confidence score measure between the recognised phone and set of possible phone variations generated by the system. This confidence score was used to determine the type of feedback given to users. Another example is a CAPL system for Cantonese speakers learning English [3] which used a set of context-sensitive phonological rules to take language transfer effects into account where non-native speakers tend to substitute phones in the L2 speech with phones in their primary language (L1).

Several studies apply some predefined threshold in separating correct pronunciation and mispronunciation. The study of CAPL for German speakers learning Mandarin [4] applies some threshold for phone/tone-level error detection, while [5] applies some predefined threshold on a likelihood-based "Goodness of Pronunciation" (GOP) measure on phone-level pronunciation. Moreover, the work in [6] uses human judgements to obtain data for determining the parameters of the system in order to define the threshold. Human judgements are also utilised in the sentence-level pronunciation scoring in [7] as the references to train a semi-supervised machine learning algorithm for constructing a ranking model automatically.

The method proposed in this paper combines a machine learning approach and automatic speech recognition (ASR) in a unique way, such that no rules, thresholds and human judgements are required in both training and testing. Human judgements are involved in the evaluation task, however. This means that most of the experiment can be carried out automatically by the system. Although the system presented in this paper is used for learning English pronunciation, the method can be applied to pronunciation learning of other languages.

This work is an extension of previous work [8] which explores the use of a three-difficulty-level (3DL) speech recogniser for pronunciation learning: novice, acceptable, and native. Each level is defined using n-gram phone models (tri-, bi- and uni-gram) in tandem with specific broad phonetic groups (BPGs) [9] based on underspecification to allow permissible phone substitutions. For instance, a novice difficulty level utilises a tri-gram phone model in tandem with a large set of BPGs in comparison to a native difficulty level that uses a uni-gram phone model with a small set of BPGs. The fact that more context is taken into account in the n-gram phone model, coupled with a larger set of BPGs, will positively help the recognition process. The ASR outputs obtained are then combined with the pronunciation score (subsection III-B) to investigate if the independently computed pronunciation score could predict pronunciation categories (obvious, minor, or no pronunciation error) without requiring the definition of some threshold. Moreover, this proposed combination also allows finer classification of pronunciation categories compared to the post-processed ASR outputs (correct or incorrectly recognised) of the previous work. Finally, a subjective evaluation is carried out in order to investigate the level of agreement between the predictions of the system and human judgements.

II. Identification of Pronunciation Error Using Broad Phonetic Groups (BPGs)

This section outlines how previous work [8] is processed to be better utilised. Figure 1 illustrates the 3DL phone recogniser corresponding to novice, acceptable, and native. The phone notation used in this paper is ARPABET.

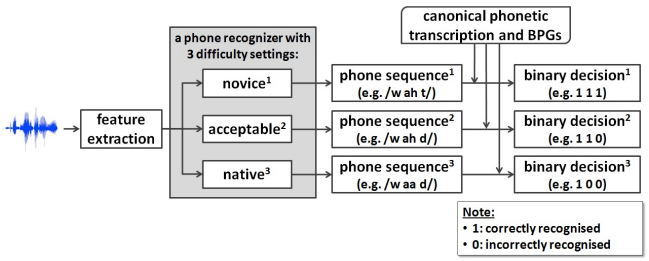


Fig. 1. The three-difficulty-level phone recogniser

The speech input in Figure 1 comes from the training set. Each speech utterance is recognised by a phone recogniser with three difficulty levels: novice, acceptable, and native. These difficulty levels determine the type of language model (n-gram) and the size of BPGs used. The quantity of BPGs for novice is greater than that at a higher difficult level. The outputs generated, which are in the form of phone sequences, are compared against canonical phonological transcriptions and difficulty-related BPGs. Depending on the difficulty level, if the canonical and recognised phones are the same or belong to the same predefined BPG at the same time interval, then the recognised phone is assumed to be correct. A label is assigned to the phone based on this decision. If the decision says the phone is correct, then label 1 is assigned to the phone; otherwise, label 0 is assigned. This means the decision made with respect to pronunciation category is only either correct or incorrect pronunciation. To obtain finer classification, the binary sequences are taken into account in tandem with pronunciation scores, which will be explored in Section III.

III. Combination of Pronunciation Scores and Binary Pronunciation Categories for Finer Classification

This section presents the overview of the system (III-A), the procedure to compute the pronunciation score (III-B), and the proposed classification method (III-C) to generate the finer three pronunciation error categories: obvious, minor, or no pronunciation error.

A. System Description

There are two steps involved in the system: training and testing, which are illustrated in Figure 2. For training, two attributes are used: the phone and its corresponding pronunciation score (III-B). Each pair of attributes has a class determining the pronunciation category of the phone. This class is extracted from the 3DL recogniser outputs. The detail on how to transform the binary decision of each output, as

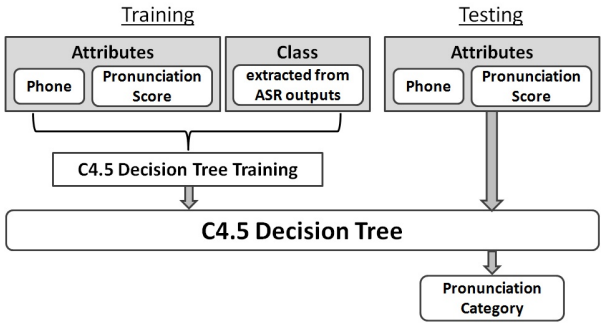


Fig. 2. The proposed system description

explained in Section II, into three classes representing the three pronunciation categories is described in Subsection III-C.

Once the training data is ready, C4.5 decision tree [10] training can be performed. In testing, speech input is preprocessed in the same way as the speech files in the training set in order to get its attributes. Then, these attributes are tested with the C4.5 decision tree model (Subsection III-C) to obtain the pronunciation category for each phone.

B. Pronunciation Score

The approach used to compute the pronunciation score is illustrated in Figure 3. It involves two types of ASR procedures: forced alignment and grammar-based ASR. Both use acoustic models trained on native speech data, which is English in this case. Forced alignment is implemented to determine segment boundaries (i.e. phone boundaries). Grammar-based ASR is used to recognise phones from the speech input by relying on acoustic models only, without being affected by language models (e.g. n-gram language models). The grammar in this case is not a rule-based phonotactics, but a list of 59 English phones from which the system can select without any constraint.

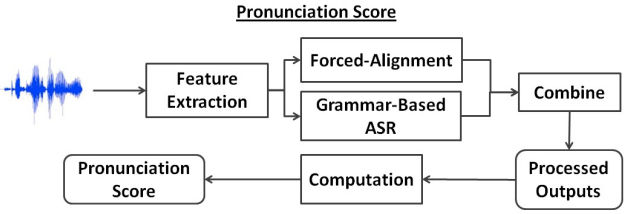


Fig. 3. Pronunciation score computation

Both forced alignment and grammar-based ASR produce files containing the phone transcription along with temporal information and acoustic likelihood for each phone. The two output files are then combined as illustrated in Table I and explained in [11].

The outputs from the forced alignment and grammar-based ASR are combined based on the temporal information of the segment boundary defined by the forced-alignment output. For instance, phone */g/* occurs from 1.31s to 1.43s in the forced-alignment output. Within the same time interval, phone */kcl/*,

TABLE I
FORCED-ALIGNMENT AND GRAMMAR-BASED (GB) AUTOMATIC SPEECH RECOGNITION (ASR) OUTPUTS AND THE COMBINED FILE

Forced-Alignment			**GB-ASR**			**Combined**			
start	end	ph.1	start	end	ph.2	start	end	ph.1	ph.2
1.24	1.31	v	1.24	1.31	f	1.24	1.31	v	f
1.31	1.43	g	1.31	1.37	kcl	1.31	1.37	g	kcl
			1.37	1.41	k	1.37	1.41	g	k
			1.41	1.46	iy	1.41	1.43	g	iy

/k/, and */iy/* are recognised by the grammar-based ASR. Thus, for combination purposes, the time interval from 1.31s to 1.43s is split based on the temporal information of both forced-aligned and recognised phones. The output can be seen in the third column of Table I (labeled "combined"). Once the combination process is done, the pronunciation score (PS) of phone p (i.e. phone */g/*) given the corresponding acoustic segment O_p (vector of acoustic features) is determined by Equation 1.

$$PS(p) = \frac{AC(O_p \mid p)}{avg_{q \in Q} AC(O_p \mid q)} \quad (1)$$

The numerator is the acoustic likelihood per frame (AC) of the acoustic segment O_p given p as its phone, which is computed from the forced alignment. The denominator represents the average of the acoustic likelihoods per frame of each phone q in the phone set Q, which occurs within the same time interval as phone p. In this example, */g/* would be p and */kcl/*, */k/*, and */iy/* would be the members of Q.

C. The Proposed Classification Method

In order to develop the finer classification for pronunciation categories, the binary decisions extracted from the 3DL phone recogniser (Figure 1) are processed further to be used as classes in training the C4.5 decision tree (Figure 2). Classes in this context describe the pronunciation categories to which each training sample belongs: obvious pronunciation error (class 1), minor pronunciation error (class 2), or no pronunciation error (class 3). Table II illustrates how the binary decisions determine the classes.

TABLE II
PRONUNCIATION CLASSES DETERMINED BY THREE-DIFFICULTY-LEVEL PHONE RECOGNISER

Class	**Error Type**	**Difficulty Level**		
		Novice	**Acceptable**	**Native**
1	obvious	✘	✘	✘
		✓	✘	✘
		✘	✓	✘
2	minor	✓	✓	✘
3	no	-	-	✓

The check mark in Table II indicates that the phone is recognised correctly, the cross mark indicates that it is not, and the hyphen indicates "don't care", which means ignore the binary decision whether the phone is correctly recognised or not. For each phone, if it is recognised by the phone recogniser with native level, then it is labeled with class 3. If it is recognised by the phone recogniser with both novice and acceptable levels, then it is labeled with class 2. Otherwise, it is labeled with class 1.

The next step is to train a C4.5 decision tree using two attributes: a phone and its pronunciation score, along with its class. The algorithm starts by calculating the normalised information gain for each attribute. The attribute with the highest gain is selected to make the decision. This process iterates until a complete decision tree is built. The training process was carried out in Weka Data Mining Toolkit [12] with a confidence factor, used for pruning, of 0.25 and the minimum number of instances per leaf was 2. In testing, the system is able to assign the pronunciation error class to each phone by using only those two attributes without having to run the 3DL recogniser, which is more efficient.

IV. EXPERIMENTAL SETUP

This section presents the experimental setup for evaluating the pronunciation classification. The following three subsections describe the data used, the design of the subjective evaluation, and the procedure to generate the syllable-level pronunciation classes from the phone-level system judgements.

A. Data

Speech from nine speakers was used in the experiments, seven males and two females, from eight different countries: Ireland (two speakers), Portugal, Indonesia, Pakistan, Netherlands, Hungary, Nigeria, and Syria. The utterances spoken were common expressions such as "How do I get to the airport?". There are 30 distinct expressions in total.

The recording was carried out in a quiet room at a sampling frequency of 16 KHz (16 bits). The speech data from the Syrian speaker, the Dutch speaker, and one Irish speaker was used for testing and the remainder used for training the decision tree. The reason to include native English speakers for testing was to investigate if the system is able to perform well for the speech being spoken by native speakers. The speech of native speakers is clearly expected to have less pronunciation errors than that of the non-native speakers, although of course regional variation may play a role. The acoustic model used in the phone recognisers (the 3DL ASR, forced alignment, and grammar-based ASR) were trained on native English speech data (TIMIT [13]).

As TIMIT was used to train the acoustic models, it is necessary to investigate the accuracy of the 3DL ASR given some speech files from the same corpus (i.e. TIMIT test set). The accuracies of the recogniser given 35 American English speech files are 85.8%, 82.7%, and 73.8% for novice, acceptable, and native levels, respectively. These numbers represent the percentages of phones that are decided to be correct by the recogniser. Note that even though the recogniser receives speech inputs with the same accent and recording environment as those of the set used for training the acoustic models, it is unlikely to obtain a perfect accuracy.

B. Subjective Evaluation

The subjective evaluation was presented through a webpage. Figure 4 presents a partial snapshot of the evaluation page. The participants for the evaluation were English native speakers who are not pronunciation experts. They performed the judgements based on their perception of whether pronunciation errors occur within the speech segments they heard. If the participants think there is a pronunciation error within the speech utterance, then they are only required to select the syllables that contain errors without having to identify if they are obvious or minor errors. This means the pronunciation categories for the subjective evaluation purposes are narrowed down to two: there *is* or *is not* pronunciation error. The reason for this decision was based on several trials where most participants were confused in distinguishing obvious and minor errors. This confusion could lead to frustation, which is likely to cause errors. Thus, narrowing the pronunciation categories down to two makes the subjective evaluation easier and less time consuming for the participants.

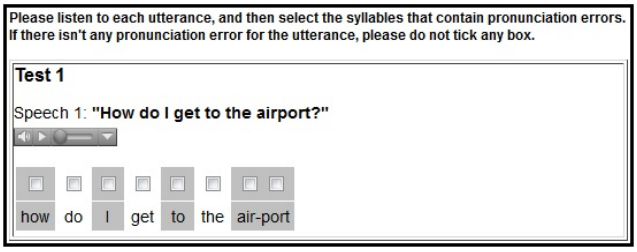


Fig. 4. A snapshot of the subjective evaluation page

Considering the time required by the participants to perform this evaluation, only 15 speech utterances were presented to be judged. Each of them was repeated once. Thus, in total there were 30 speech utterances. The repetition was taken into account in order to investigate the internal consistency of the participants in making their judgements.

C. Generating the Syllable-Level Pronunciation Classes of the System Judgements

As mentioned in subsection IV-B, judgements were made on the syllable level. Thus, in order to measure the level of agreement between the opinions of the participants and the output of the system, the phone-level pronunciation classes produced by the system are processed further to obtain the pronunciation classes on the syllable level. The procedure underlying the decision of the pronunciation class for each syllable is illustrated in Figure 5.

Given a syllable s consisting of n phones, if there is at least one phone within s that belongs to class 1 (obvious pronunciation error), then s is classified as having an obvious pronunciation error; otherwise, the pronunciation class of s is determined by the number of occurences of the remaining classes. If the number of occurences of class 2 (minor pronunciation error) is larger than or equal to that of class 3 (no pronunciation error), then s is considered to have minor

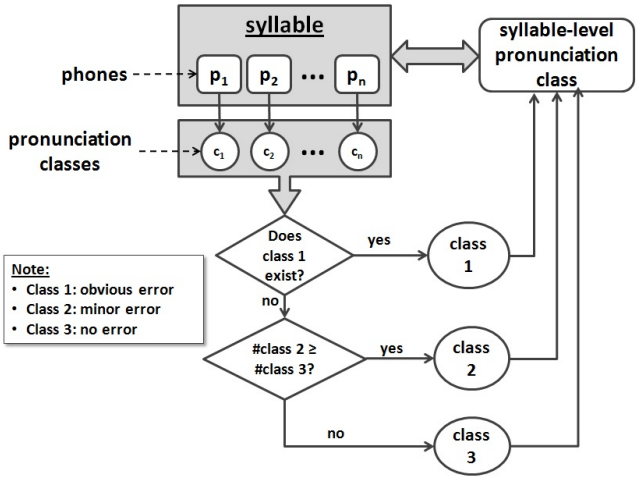


Fig. 5. Rule of determining pronunciation error classes of a syllable based on its constituent phones

pronunciation error. Otherwise, it is considered to have no pronunciation error. This rule is applied to determine the pronunciation classes for all syllables in the evaluation set.

V. Results and Analysis

Twelve participants took part in the evaluation task. Table III presents the levels of agreements between the participants and the system judgements.

TABLE III
Agreement Between System and Human

		Human Judgement	
		pronunciation error?	
		Yes	No
System Judgement	**Obvious (class 1)**	**51.1%** (135/264)	48.9%
	Minor (class 2)	**19.2%** (76/396)	80.8%
	No (class 3)	8%	**92%** (629/684)
Overall Agreement = 62.5%			

The percentages highlighted in bold in Table III represent the levels of agreements between the system and human judgement, which are 51.1%, 19.2%, and 92% for pronunciation class 1, 2, and 3, respectively. The percentage represents the portion of occurrences of a particular class generated by the system that is judged by the humans as its corresponding class. Note that the participants did not distinguish between obvious and minor errors, rather distinguished whether there is error or not. Thus, when the system decides that a syllable has either an obvious or minor error and the participant says that it has an error, then this contributes to the level of agreement. For instance in Table III, 51.1% agreement between the system (obvious error) and human judgements (error) means that 135 of 264 obvious errors identified by the system are judged as errors by humans. Thus, the counts in Table III are based on

the situation where the 2-class human judgements are mapped into the 3-fold classes.

From Table III, it seems that the system is able to detect pronunciations with no errors extremely well, while for the ones that have errors, the agreements between the system and humans are weak, especially class 2 (minor error). The overall level of agreement results in 62.5%.

Additional to the percentage of agreement, false positive and false negative rates [14] were also measured. False positive is the circumstance when the system outputs that there is no pronunciation error while the evaluator says the opposite. Otherwise when the system outputs that there is a pronunciation error while the evaluator finds none, it is called false negative. The false positive and false negative rates obtained in this experiment are presented in Table IV.

TABLE IV
FALSE POSITIVE AND FALSE NEGATIVE RATES

False Positive	**False Negative Rate**	
Rate	**class 1**	**class 2**
20.7%	17%	33.7%

The false positive rate obtained here (20.7%) is not a significant percentage (compared to the results obtained in [3]). Moreover, the false negative rate of class 2 (33.7%) is higher than that of class 1 (17%), as expected. Assuming that human judgements are correct, the system performs better at detecting class 2 rather than class 1.

As mentioned in subsection IV-B, the speech utterances used in the subjective evaluation were repeated once randomly to investigate the consistency of the participants in making judgements. In this evaluation, for each participant, the internal inconsistency is 24%, on average. This means that 24% of the time, the participants made a different decision on the same utterance. As the participants already display inconsistency, it is understandable that in the case of pronunciation errors, especially the minor ones, an overall agreement of 100% will never be achieved.

VI. CONCLUSION AND FUTURE WORK

This paper presented a method of classifying pronunciation classes into finer categories: obvious, minor, or no pronunciation error. By combining the processed outputs of the 3DL phone recogniser and the pronunciation score into a C4.5 decision tree, the system is able to classify the pronunciation into the three categories aforementioned. Based on the experiments carried out, the system performed extremely well in detecting correct pronunciations. However, the performance of the system seems weak in detecting pronunciation errors, especially minor errors (class 2), assuming the human judgements are correct. There are two possibilities with respect to this issue, either the system performs poorly in such cases or the participants involved in the subjective evaluation have low strictness due to the fact that they are not English pronunciation experts, but regular English speakers. This issue needs to be investigated further. Moreover, incorporating other corpora will be taken into account in order to build the acoustic model.

Apart from the outcomes of the evaluation, the proposed method has two advantages. Firstly, it does not require any rule, threshold, and human judgement in training the model and testing. Secondly, it can be applied for the pronunciation learning of other languages as long as the acoustic models used in the 3DL phone recogniser and the two phone recognisers involved in computing the pronunciation score cover the features of all phones corresponding to the target language(s).

Future work will seek to investigate the proper and appropriate types of feedback presented to the users so that the system is useful in supporting their learning experience. Furthermore, the level of usefulness of the system is necessary to evaluate.

ACKNOWLEDGMENT

This research is supported by the Science Foundation Ireland (Grant 07/CE/I1142) as part of the Centre for Next Generation Localisation (www.cngl.ie) at University College Dublin (UCD). The opinions, findings, and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of Science Foundation Ireland.

REFERENCES

[1] M. Eskenazi, "An overview of spoken language technology for education," Speech Communication, vol. 51, pp. 832-844, 2009.
[2] S. M. Abdou, S. E. Hamid, M. Rashwan, A. Samir, O. Abd-Elhamid, M. Shahin, and W. Nazih, "Computer aided pronunciation learning system using speech recognition techniques," Proc. Interspeech, 2006.
[3] W. K. Lo, S. Zhang, and H. Meng, "Automatic derivation of phonological rules for mispronunciation detection in a computer-assisted pronunciation training system," Proc. Interspeech, 2010, pp. 765-768.
[4] H. Hussein, S. Wei, H. Mixdorff, D. Külls, S. Gong, and G. Hu, "Development of a computer-aided language learning system for Mandarin - tone Recognition and pronunciation error detection," Proc. Speech Prosody, Chicago, Illinois, May 2010.
[5] S. M. Witt and S. J. Young, "Phone-level pronunciation scoring and assessment for interactive language learning," Speech Communication, vol. 30, pp. 95-108, 2000.
[6] S. E. Hamid, O. Abdel-Hamid, and M. Rashwan, "Performance tuning and system evaluation for computer aided pronunciation learning," 2009.
[7] L. Y. Chen and J. S. R. Jang, "Automatic pronunciation scoring using learning to rank and DP-based score segmentation," Proc. Interspeech, 2010, pp. 761-764.
[8] M. Kane, J. P. Cabral, A. Zahra, and J. Carson-Berndsen, "Introducing difficulty-levels in pronunciation learning," Proc. SLATE, Venice, 2011.
[9] P. Scanlon, D. Ellis, and R. Reilly, "Using broad phonetic group experts for improved speech recognition," Proc. IEEE Transactions on Audio, Speech and Language Processing, vol. 15, no.3, Pages 803812, 2007.
[10] J. R. Quinlan, "C4.5: Programs for machine learning," Morgan Kaufmann, San Mateo, CA, 1993.
[11] M. Kane and J. Carson-Berndsen, "Multiple source phoneme recognition aided by articulatory features," Trends in Applied Intelligent Systems: Lecture Notes in Artificial Intelligence (LNAI 6704 part 2) - IEA/AIE, 2011.
[12] I. H. Witten, E. Frank, L. Trigg, M. Hall, G. Holmes, and S. J. Cunningham, "Weka: Practical machine learning tools and techniques with Java implementations," H. Kasabov and K. Ko, eds., ICONIP/ANZIIS/ANNES'99 International Workshop, Dunedin, 1999.
[13] J. Garofolo, L. Lamel, W. Fisher, J. Fiscus, D. Pallett, and N. Dahlgren, 1993. The DARPA TIMIT Acoustic-Phonetic Continuous Speech Corpus CDROM.
[14] T. M. Mitchell, Machine Learning. international ed., xvii, 414p, London: McGraw-Hill, 1997.

Automatic Detection of [θ] Pronunciation Errors for Chinese Learners of English

Keelan Evanini and Becky Huang
Educational Testing Service
Princeton, NJ 08541

***Abstract*—This study examines the types of errors produced by Chinese learners of English when attempting to pronounce [θ] in reading passages and presents a system for automatically detecting these pronunciation errors. The system achieves an accuracy of 79.8%, compared to the inter-annotator exact agreement rate of 83.1%. In addition, speaker-level scores based on the total number of correct productions of [θ] made by each speaker are generated from both the human and machine error annotations, and these are shown to have a strong correlation with each other (0.797).**

I. Introduction

The English voiceless interdental fricative, [θ], is a difficult sound for many non-native speakers to master, and is quite rare cross-linguistically. The difficulty of acquiring a native-like pronunciation of [θ] can be shown by the fact that many long-term learners of English residing in English-speaking countries continue to make this error. In the case of fossilized errors like this, explicit instruction and intense individual practice with the target phone is required to help the language learner achieve a native-like pronunciation. This is thus an ideal application for an automated pronunciation error detection system, since the level of individual attention required to change a learner's behavior would be more than is possible in a typical instructional environment.

In this study, we focus on the specific L1 background of Mandarin Chinese. Studies have shown that Mandarin learners of English have difficulty acquiring [θ] in English and typically use the phone [s] as a substitution [1], [2]. This study focuses on a set of adult speakers of Mandarin Chinese who have been residing in the United States for extended periods, and examines the performance of an automated system for detecting [θ] pronunciation errors on this group of speakers.

There have been many prior approaches to automated pronunciation error detection. The most widespread method uses confidence scores obtained from the ASR system, as in [3] and [4]. Other, more recent approaches, have investigated the use of classifiers based on spectral characteristics, as in [5], or a combination of both approaches, as in [6]. In this study, we adopt a simple approach based on modifying the pronunciation dictionary to contain pronunciations with errors and using forced alignment to select the variant with the highest acoustic score, as in [7]. This approach was used since it can be done relatively easily using open-source capabilities; thus, it has the potential to be used in a wide variety of Computer-Assisted Language Learning applications.

This paper is organized as follows: first, Section II presents the materials that were used in collecting the data for this study and the characteristics of the speakers; Section III describes the annotation procedure that was followed to produce phonetic transcriptions for the learners' tokens of [θ]; Section IV describes the methodology that was used to automatically detect [θ] pronunciation errors; Section V presents analyses of the error detection results; finally, Section VI summarizes the study and describes future related work.

II. Data Collection

This study used three isolated sentences and a paragraph as stimuli. The three sentences were designed by [8] for a foreign accent rating experiment, and each contains one word with the target phone, [θ]. These three sentences are listed below, with the word containing the target phone in bold:

1) *Ron set a **thick** rug in the sun.*
2) *You should **thank** Sam for the food.*
3) *It is fun to play chess **with** a rook.*

In addition, the study also used the *Stella* paragraph [9], a reading passage that is commonly used in accent research. The reading passage contained five instances of the target phone:

> *Please call Stella. Ask her to bring these **things with** her from the store: six spoons of fresh snow peas, five **thick** slabs of blue cheese, and maybe a snack for her brother Bob. We also need a small plastic snake and a big toy frog for the kids. She can scoop these **things** into **three** red bags, and we will go meet her Wednesday at the train station.*

Thus, there were a total of 8 tokens containing the target phone [θ] in this study, and 5 lexical types (*thick*, *with*, and *things* each appeared twice).

36 native speakers of Mandarin (13 male, 23 female) who are long-term residents in the USA participated in the study. All participants arrived in the USA after the age of 18, and all have lived in the USA for a minimum of 7 years (range = 7 - 26; mean = 10, st. dev. = 4). The participants read each of the three sentences and the paragraph out loud twice. Their speech was recorded using a headset microphone (Shure SM 10A) and Audacity (v. 1.2.5) in a quiet location; the audio files were sampled at 16 kHz and saved as uncompressed WAV files. The total number of non-native productions of [θ] investigated in this study is thus 576 (36 participants * 8 tokens * 2 repetitions).

In addition, 22 native speakers of American English (10 male, 12 female) were included in the study as a control group. They read the same materials as the non-native speakers and were recorded under the same conditions. The total number of native productions of [θ] in this study is 352 (22 * 8 * 2).

III. Annotation

Both of the authors of this paper[1] independently listened to the recordings of the sentences and paragraphs produced by the non-native speakers and provided phonetic transcriptions for each of the 576 instances of the target phone. In addition to using perceptual cues to produce the transcriptions, the annotators also incorporated information from the waveform and the spectrogram when the perceptual cues were ambiguous.

The phonetic transcription process revealed that the participants in this study produced a wide range of substitutions to replace the target phone [θ]. In addition to the expected variant [s], the following English phones were also used occasionally as substitutes: [d], [ð], [f], [t], [tʃ], and [z]. Finally, two further sounds which are somewhat harder to characterize were occasionally used as substitutes for [θ]. First, some speakers produced a sound which clearly started out as [s], but then ended with an interdental release (either a stop or a fricative). In these cases, it appeared that the speaker first substituted [s] for [θ], but then became conscious of this mispronunciation and attempted a strategy for correcting it. These tokens are labeled as [sθ], and they only occurred word-initially (i.e. not in the word *with* in this data set). It is likely that their frequency would be much lower in unmonitored spontaneous speech. The other variant that is problematic to describe sounds like an interdental stop. In these cases, there is no sustained portion of aperiodic noise that would be characteristic of a fricative, but the place of articulation sounds quite different from a canonical alveolar stop, [t]. We labeled these interdental stops as [t̪].[2]

TABLE I
Confusion Matrix for Human Annotations

		Annotator BH										
		[d]	[ð]	[f]	[s]	[sθ]	[ʃ]	[t]	[t̪]	[tʃ]	[θ]	[z]
Annotator KE	[d]											
	[ð]		1								3	2
	[f]										1	
	[s]				126	7	2			1	30	
	[sθ]				3	10			1		6	
	[ʃ]											
	[t]		1					3				
	[t̪]	1				1		2	18		6	
	[tʃ]									1		
	[θ]	1	2		28	16		4	35		250	
	[z]				1						1	10

Table I presents the confusion matrix for the two annotators. The inter-annotator exact agreement rate was 72.7% with $\kappa = 0.55$ (the total number of phonetic symbols used in the annotation task was 11). After the two annotators completed annotating the tokens independently, all cases of disagreement were adjudicated by the two annotators together. For the adjudication round, the annotators did not have access to their original annotations, but listened to each audio sample and examined the spectrogram together to come to an agreement. Table II presents the distribution of the annotations on the 576 tokens after adjudication.

TABLE II
Distribution of Annotations after Adjudication

Annotation	Frequency	Annotation	Frequency
[θ]	289	[ð]	4
[s]	166	[tʃ]	2
[t̪]	69	[ʃ]	1
[sθ]	24	[f]	1
[z]	12	[d]	1
[t]	7		

IV. Methodology

To detect pronunciation errors in the non-native productions of [θ], we used the Penn Phonetics Lab Forced Aligner [10]. This open-source forced alignment toolkit is based on HTK [11] and contains monophone acoustic models trained on 25.5 hours of native speech. To model the most frequent type of [θ] error produced by the non-native speakers in this data set, we modified the pronunciation dictionary to include additional pronunciations of the target words containing the phone S instead of TH.[3] For example, the modified dictionary contained the following two entries for the word *thick*:

THICK TH IH1 K
THICK S IH1 K

Then, the recorded utterances were subjected to forced alignment with the stimulus texts using this modified pronunciation dictionary; no modifications were made to the transcriptions to account for disfluencies or reading errors. During the process of forced alignment, the system selects the pronunciation from the dictionary containing either TH or S based on which phone's model is the closest match to the acoustic features. When the forced aligner outputs TH for one of the tokens, this is categorized as a correct pronunciation of the target phone [θ]; alternatively, an output of S for a given token is categorized as a pronunciation error. In the following section, we will compare these machine classifications with the gold standard labels provided by the human annotators.

V. Results

This section presents the results of [θ] pronunciation error detection. However, evaluating the results is not straightforward, due to the fact that the non-native speakers produced a large number of different pronunciation variants for [θ], but

[1]The first author is a native speaker of English with no knowledge of Mandarin and the second author is a native speaker of Mandarin.

[2]This variant is also relatively common in speech produced by native speakers–it occurred several times in a random sample of the responses from the native speaker control group.

[3]Additional experiments were conducted with multiple pronunciation errors in addition to [s] included in the dictionary; however, this approach decreased the performance of the system.

the system only predicts two phones ([θ] and [s]). In Section V-A we first present the results on two different subsets that contain only the two most frequent variants: [θ] and [s]. Then, in Section V-B we present the results for all tokens in order to estimate the performance in an actual application where a decision must be made about every token.

A. Tokens Annotated as [s] and [θ]

As described in Section III, the speakers in this study substituted a wide range of pronunciation variants for the target [θ]. Since the pronunciation error detection system is only designed to classify pronunciations as either the target [θ] or the variant [s], and since the human annotations included several phones in addition to these two, the evaluation of the system's performance is not a straightforward task. Therefore, we first evaluate its performance on the following two subsets of the data containing only adjudicated annotations of [θ] or [s] for which the evaluation task is more straightforward:

- ADJ_TH_S: This subset contains the 455 tokens that received an adjudicated annotation of either [θ] or [s].
- ANN_TH_S: This subset contains the 429 tokens that received an annotation of [θ] or [s] from both annotators during the round of independent annotation (this set is a subset of ADJ_TH_S). This subset was thus intended to only include the tokens which were unambiguous instances of voiceless fricatives so that the system's performance could be examined on the most prototypical cases.

For these two subsets, a direct comparison between the adjudicated annotation and the phone output by the forced aligner is thus possible. Table III presents the inter-annotator agreement and the automatic error detection results for these two subsets of tokens. The inter-annotator agreement statistics were computed by comparing the two sets of independent annotations (before adjudication). The machine detection accuracy results were calculated by comparing the output of the forced aligner with the gold standard adjudicated annotations. The precision and recall values show how well the machine system detected errors; that is, these values were computed with respect to the [s] category.

TABLE III
[θ] ERROR DETECTION RESULTS ON TWO SUBSETS

Experiment	N	Task	% Agree	κ	Prec.	Rec.
ADJ_TH_S	455	human	0.826	0.65	–	–
		machine	0.813	0.60	0.748	0.734
ANN_TH_S	429	human	0.876	0.73	–	–
		machine	0.811	0.59	0.740	0.735

As Table III shows, the performance of the error detection system was similar to the human-human agreement for the ADJ_TH_S subset, but the human-human agreement was higher on the ANN_TH_S subset. The higher human agreement on the ANN_TH_S subset can be attributed to the fact that the tokens included in it were likely more distinct perceptually, since they all received an initial annotation of either [θ] or [s].

B. All Tokens

In this section, we present the error detection results on all of the tokens in the data set. As discussed above, the evaluation of the system's performance on this task is less straightforward, since the set of adjudicated annotations contains more phones than were used by the error detection system. So, it is first necessary to merge the adjudicated annotations into two categories that correspond to the two phones produced by the system. Therefore, the human annotations were divided into a group consisting of *correct* tokens and *errors*. The *correct* category included the target phone [θ] along with the two pronunciation variants that native speakers may also produce: [t̪] and [ð].[4] The *error* category included all other variants produced by the participants, none of which would be expected from a native speaker: [s], [sθ], [z], [tʃ], [ʃ], [f], [d]. The two phones output by the forced aligner corresponded in a straightforward manner to these two categories: TH corresponded to the *correct* category and S corresponded to the *error* category. This experiment in which the token labels were merged to create a binary distinction will be referred to as ALL_BINARY below.

Table IV first presents the human-human agreement results for all of the tokens before they were merged (ALL). Then, the results are presented using the annotations that were merged into the *correct* and *error* categories (ALL_BINARY).

TABLE IV
[θ] ERROR DETECTION RESULTS ON ALL TOKENS

Experiment	N	Task	% Agree	κ	Prec.	Rec.
ALL	576	human	0.727	0.55	–	–
ALL_BINARY	576	human	0.831	0.64	–	–
		machine	0.798	0.56	0.766	0.658

As Table IV shows, the accuracy rate achieved by the error detection system on all of the tokens (0.798) is only 3.3% lower than the exact agreement rate achieved by the two human annotators (0.831). However, the κ value is 8% lower; furthermore, the recall of the error detection system declines substantially when all tokens are included. This indicates that the performance of the error detection system suffered on the categories that were labeled as *error* in the ALL_BINARY set but were excluded from the ANN_TH_S and ADJ_TH_S sets. This higher incidence of false negatives can be shown in the three highest frequency annotations in the *error* category after [s]: the error detection system only predicted 50% of the [sθ] variants as errors (12 / 24), 50% of the [z] variants (6 / 12), and 0% of the [t] variants (0 / 7).

C. Native Speaker Results

As an additional test of the validity of the automated error detection system, it was also applied to the set of native speaker responses. No annotation was conducted for this experiment, since it was assumed that all native speaker tokens

[4]The 4 tokens with the [ð] variant produced by the non-native participants occured word-finally before a vowel in the function word *with*, an environment in which native speakers may also produce the voiced [ð].

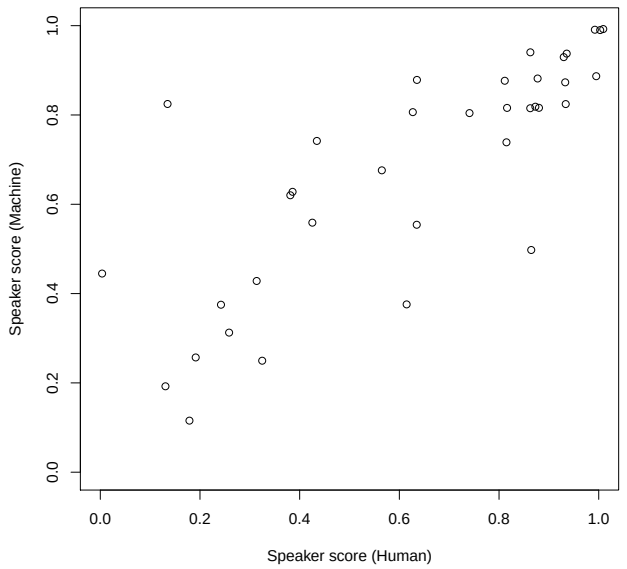
would fall into the *correct* category (i.e., would be one of the following three variants: [θ], [t̪], or [ð]). Out of the 352 tokens in this data set, 337 were classified by the system as TH and 15 were classified as S. Asssuming that none of the tokens should have been classified as an *error*, this amounts to a 4.3% false positive rate for this data set.

D. Speaker-level Results

In order to evaluate the usefulness of the automated error detection system for diagnostic or placement purposes, speaker-level [θ] scores were calculated for each non-native participant in the study based on the percent of *correct* tokens they produced. Scores were produced based on both the human annotations and the machine error detection results by dividing the total number of tokens labeled as *correct* in the ALL_BINARY condition by the total number of tokens produced by the speaker (16). This value thus provides a holistic speaker-level score for each participant's proficiency in producing the phone [θ].

Figure 1 shows that the speaker-level [θ] scores produced by the automated system correspond well with the scores from the human annotations.[5] The Pearson correlation between the speaker-level [θ] scores based on human annotations and those based on machine predictions was 0.797 ($p < 0.001$).

Fig. 1. Speaker-level scores for % of tokens produced correctly



As Figure 1 shows, the speaker with the largest divergence between the human and machine [θ] scores had a machine score of 0.813 (13 / 16 *correct*) and a human score of 0.125 (2 / 16 *correct*). One possible explanation for the poor performance of the error detection system on this speaker is the fact that the audio quality of all of the responses for this speaker was severely degraded by the presence of a constant source of static in the signal. This additional noise in the signal may have caused the spectral characteristics of the speaker's productions of the variant [s] to be more similar to the forced aligner's models for [θ], thus causing a large number of false negatives for this speaker.

[5]The values for points in the figure that are represented by multiple speaker-level scores, such as (1.0, 1.0), were slightly perturbed so that all points could be seen.

VI. Conclusion

In this study we have demonstrated that a simple [θ] pronunciation error detection system based on forced alignment with a modified pronunciation dictionary and open-source native-speaker acoustic models achieves a level of performance that is close to the inter-annotator agreement rate for this data set. In addition, we showed that a speaker-level [θ] production accuracy scores based on the automated error detections has a strong correlation with the scores based on human annotations. These results indicate that the error detection system can provide valid feedback to Chinese learners of English in terms of their production accuracy.

The process of pronunciation error annotation and adjudication used in this study provides a rich foundation of knowledge on which analyses of the performance of an error detection system can be based. The fact that several pronunciation variants occurred that were not expected based on the literature suggests that researchers should always use real learner corpora (not artificial errors) and provide detailed transcriptions of their data so they can fully evaluate the performance of their systems. Future research will incorporate more state-of-the-art error detection techniques and apply them to detecting [θ] errors from speakers with a variety of first languages.

References

[1] D. V. Rau, H.-H. A. Chang, and E. E. Tarone, "Think or sink: Chinese learners' acquisition of the English voiceless interdental fricative," *Language Learning*, pp. 581–621, 2009.

[2] J. Xiao and Y. Zhang, "A study of Chinese EFL learners' acquisition of English fricatives," in *Proceedings of the 16th Conference of the Pan-Pacific Association of Applied Linguistics*, 2011.

[3] S. Witt, "Use of the speech recognition in computer-assisted language learning," Ph.D. dissertation, Cambridge University, 1999.

[4] H. Franco, L. Neumeyer, M. Ramos, and H. Bratt, "Automatic detection of phone-level mispronunciation for language learning," in *Proceedings of Eurospeech*, 1999.

[5] H. Strik, K. Truong, F. de Wet, and C. Cucchiarini, "Comparing classifiers for pronunciation error detection," in *Proceedings of Interspeech*, 2007.

[6] S.-Y. Yoon, M. Hasegawa-Johnson, and R. Sproat, "Automated pronunciation scoring using confidence scoring and landmark-based svm," in *Proceedings of Interspeech*, 2009.

[7] D. Herron, W. Menzel, E. Atwell, R. Bisiani, F. Daneluzzi, R. Morton, and J. A. Schmidt, "Automatic localization and diagnosis of pronunciation errors for second-language learners of English," in *Proceedings of Eurospeech*, 1999.

[8] J. E. Flege, G. H. Yeni-Komshian, and S. Liu, "Age constraints on second-language acquisition," *Journal of Memory and Language*, vol. 41, pp. 78–104, 1999.

[9] S. Weinberg, "Speech Accent Archive," http://accent.gmu.edu, 2011.

[10] J. Yuan and M. Liberman, "Speaker identification on the SCOTUS corpus," in *Proceedings of Acoustics '08*, 2008.

[11] S. Young, G. Evermann, M. Gales, T. Hain, D. Kershaw, X. Liu, G. Moore, J. Odell, D. Ollason, D. Povey, V. Valtchev, and P. Woodland, *The HTK Book*, 2006. [Online]. Available: http://htk.eng.cam.ac.uk/docs/docs.shtml

Pronunciation Training by Extracting Articulatory Movement from Speech

Tsuneo Nitta, Silasak Manosavan, Yurie Iribe, Kouichi Katsurada
Graduate School of Engineering
Toyohashi University of Technology
Toyohashi, JAPAN
nitta@cs.tut.ac.jp

Ryoko Hayashi[1], Chunyue Zhu[2]
[1] Graduate School of Intercultural Studies
[2] School of Language and Communication
Kobe University
Kobe, JAPAN
rhayashi@kobe-u.ac.jp

***Abstract*— In this paper, we describe computer-assisted pronunciation training (CAPT) through the visualization of learner's articulatory gesture. Typical CAPT systems evaluate pronunciation by using speech recognition technology, however, they cannot indicate how the learner can correct his/her articulation. The proposed system enables the learner to study how to correct pronunciation by adjusting the articulatory organs highlighted on a screen and comparing with the correctly pronounced gesture. In the system, a multi-layer neural network (MLN) is used to convert learner's speech into the coordinate of a vocal tract using MRI data. The vocal tract area data is applied for the input of MLN and compared with articulatory features (AF) extracted from the same utterance as the MLN input. Then, a CG generation process outputs articulatory gesture using the values of the vocal tract coordinate. The comparison of the extracted CG animation from speech and the actual MRI data is investigated.**

Keywords- Interactive pronunciation training, Articulatory feature extraction, Articulatory gesture CG-generation.

I. INTRODUCTION

Computer-assisted pronunciation training (CAPT) has been introduced for language education in recent years [1], [2]. Typical CAPT systems evaluate the pronunciation of learners and point out the articulation error by using speech recognition technology [3], [4], [5]. Moreover, some of them can indicate the differences between incorrect and correct pronunciation by displaying speech waveform or 1st and 2nd formant frequencies. The learners can aware of the differences, however, they cannot correct the pronunciation only by such information unless they have sufficient knowledge in phonetics. On the other hand, some studies have introduced sagittal articulatory information by animations or video of correct gesture [6], [7], however, because these approaches do not feedback the learner's incorrect gesture at the same time, these types of articulatory feedback do not in fact help learners. The CAPT systems should guide learners how to adjust articulatory organs when correcting pronunciation error. We have studied pronunciation training based on articulatory feature extraction from speech [8], [9] that realizes visualization of learner's pronunciation error (Figure 1). We expect that the step-by-step learning process using CG animation enables a learner to study how to correct his/her pronunciation by adjusting articulatory movement highlighted on a screen and comparing with the correct one.

In the proposed system, a multi-layer neural network (MLN) is used to convert learner's speech into the coordinate of a vocal tract using MRI data [10]. The vocal tract area (VTA) data extracted from speech is applied for the input of MLN and compared with articulatory features (AFs; place of articulation and manner of articulation) extracted from the same speech as the MLN input [11]. Then, a CG generation process outputs articulatory gesture using the values of the coordinate. The comparison of the extracted CG animation from speech and the actual MRI data is investigated.

In section 2, the calculation of vocal-tract cross-section, the coordinate vector extraction, and the CG animation generation are described. In section 3, experimental results are discussed by comparing the extracted animation with MRI data. In the last section, the paper is summarized.

II. ANIMATED PRONUNCIATION GENERATION

A. *System outline*

Figure 2 shows a system outline. The system consists mainly of a VTA extractor [9] or an AF extractor [8], a coordinate vector extractor using MLN, and a CG animation generator. The coordinate vectors are acquired by transforming the VTA or AF. In this paper, we experiment in extracting vocal tract areas directly to investigate the coordinate values in articulation organs by comparing with AFs.

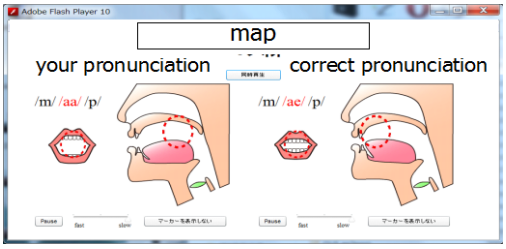


Figure 1: Animated Pronunciation: learner/ teacher.

The CG animation is generated on the basis of coordinate values extracted from the trained MLN. As a result, user's speech is input in our system, and then a CG animation is automatically generated to visualize the pronunciation movement.

B. Vocal tract area extraction

The vocal tract area A_m is determined by the following equation.

$$A_{m-1} / A_m = (1+k_m)/(1-k_m), (m = M,...,1) \quad (1)$$

Here, m is the number of cross-sections in a vocal tract, k_m is a PARCOR coefficient. PARCOR coefficients are equivalent to reflection coefficients in a lossless acoustic tube model of the vocal tract. The vocal tract area function expresses the vocal tract area from the glottis to the lips, and is related to the distance between palate and tongue. The vocal tract area is calculated with PARCOR parameters extracted from speech signals. The extracted vocal tract area (13 dimensions) is combined with the other two frames, which are three points prior to and following the current frame (VT(t, t-3), VT(t, t+3)) to form articulatory movement, and then the vocal tract area (13 × 3 dimensions) is input to the MLN.

C. Coordinate vector extraction

We apply magnetic resonance imaging (MRI) data to obtain the coordinate values of the shape of an articulatory organ. MRI machines capture images within the body by using magnetic fields and electric waves. The MRI data captured in two dimensions detail the movements of the person's tongue, larynx, and palate during utterance using a phonation-synchronized imaging [10]. The data-set of MRI and speech used here is 36 vocabulary-words uttered 192 times each by a female American-English native speaker that is the sub-set of an MRI corpus including two English native speakers (one female and one male) and two non-native speakers (one Japanese female and one Japanese male). CG animations are generated on the basis of coordinate vectors. The MLN trains the vocal tract areas as input, that are extracted from speech recorded at the MRI data collections, and the coordinate vectors of the articulatory organs acquired from the MRI images as output (Figure 3). As a result, after the input of user's speech, the coordinate vectors adjusted to the speech are extracted, and then a CG animation is generated. In this section, the extraction of the feature points on the MRI data and the method for calculating the coordinate vectors of each feature point are described.

We assigned initial feature points to the articulatory organ's shapes (tongue, palate, lips, and lower jaw) on the MRI data beforehand. The number of initial feature points was 43 (black-colored points in Figure 4). Then, we decreased the number of dimensions in the MLN in order to train the MLN effectively with a small amount of MRI data. We selected eight feature-points that are important at pronunciation training (Figure 5). The feature points used here are obtained by the following steps.

1) 10-ms speech and image segment from the MRI data are imported.

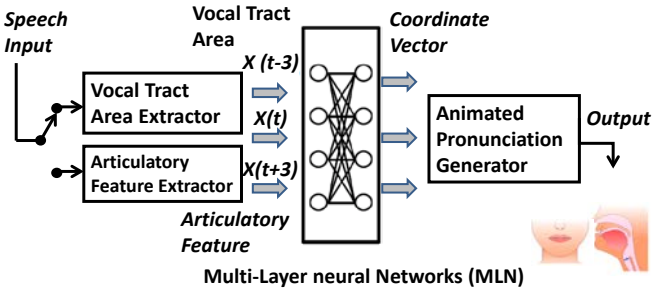


Figure 2: System Outline

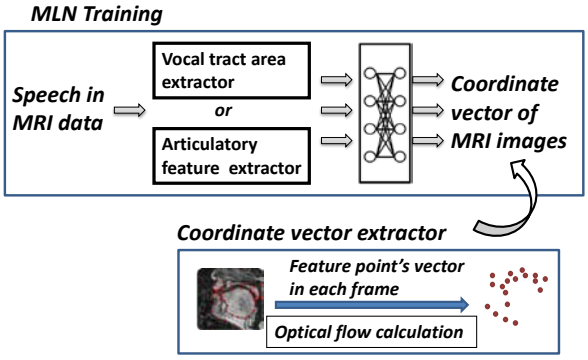


Figure 3: Coordinate vector extraction

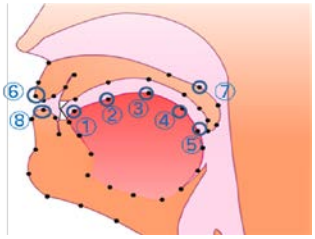


Figure 4: Feature points used in MLN training

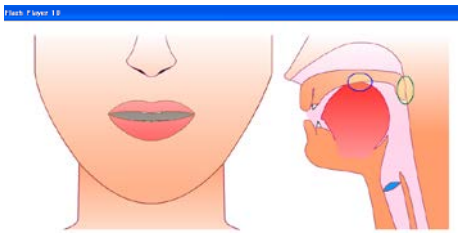


Figure 5: CG animation of "basket"

2) The coordinate value of each feature-point is extracted by calculating the optical flow in each frame. The input data of the optical flow program is a coordinate vector set at the initial feature points.

3) y-coordinate distance of each feature point is calculated for the reduction of dimensions. The x-coordinate value was fixed to the same as the initial feature- point.

The numbers of dimensions in each MLN-unit are; (a) input unit 45 (15 × 3) vocal tract areas, (b) output unit 24 (8 × 3) y-coordinate vectors.

D. CG animation generation programs

We, firstly, assigned 43 points (15 tongue points, 2 lip points, 16 palate points, and 10 lower jaw points) as the initial feature-points of the MRI image. Then, the position relations among 8 important feature-points used for training at MLN and remaining 35 feature-points are calculated. The spline curve is used to complement eight feature-points and other feature-points by keeping the position relation. The movement is drawn on the basis of y-coordinate distance, however, since this movement is often unstable, we introduce a median filter to smooth it out.

A pronunciation training system is built as a web application so that various users can access on the web. The CG animation program is implemented with Actionscript3.0 to operate on web browser with Flash Player plug-in. Figure 5 shows a screen shot of a CG animation developed in the present study.

III. EVALUATION

The correlation coefficients between the coordinate values in CG animations extracted by MLN and their corresponding values in articulatory gestures investigated on the MRI data are evaluated. When extracting coordinate vectors of a vocal tract, we compare two features as the input of MLN, namely, the vocal-tract cross-sections, or vocal tract areas, and the articulatory features (AF).

A. Experimental data and setup

The MRI data used in the evaluation was taken in a single shot, in which a female English native speaker uttered 37 English words. The data set used in the experiment is as follows.

D1: *Training data set for an AF-coordinate vector converter or a VT-coordinate vector converter: 36 short words of English speech and images included in the MRI data.*

D2: *Testing data set for an AF-coordinate vector or VT-coordinate vector converter: One word of English speech included in the MRI data (one female English native speaker).*

The MLN for the AF extractor [11] was designed using TIMIT database. Experiments are conducted by using a leave-one-out cross-validation method, that is, 36 trials are investigated.

B. Experimental results

Figure 6 shows the correlation coefficient for each phoneme. As for the averaged correlation coefficient ("all" in Figure 6) of all the phonemes, the vocal tract area was 0.83 and articulatory feature was 0.78. In addition, the correct rate of articulatory features using 2-stage MLNs [8] is 81.2% (“all” in Figure 7). These results show that; (1) comparatively high correlation coefficient is achieved in spite of a small amount of training data, (2) the vocal tract areas outperform the articulatory features in animated pronunciation evaluation.

Concerning the mapping to coordinate vectors of animated pronunciation, the vocal tract approach has higher correlation than the articulatory feature approach. However, because the vocal tract approach depend on speakers and the test was executed with only one speaker, further investigation is needed to fix the design of animated pronunciation generator. Moreover, the articulatory feature approach has another advantage that it is basically speaker-unspecific and it is expected that the articulatory feature outperforms the vocal tract area totally in speaker-independent experiments.

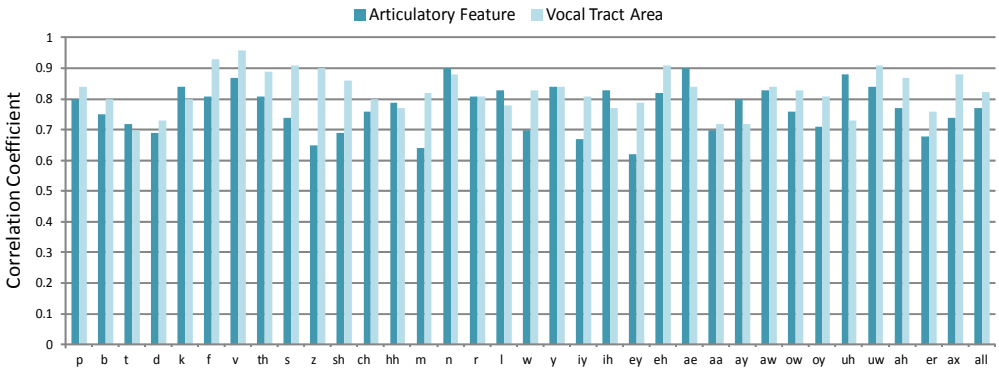


Figure 6: Correlation coefficient of each phoneme

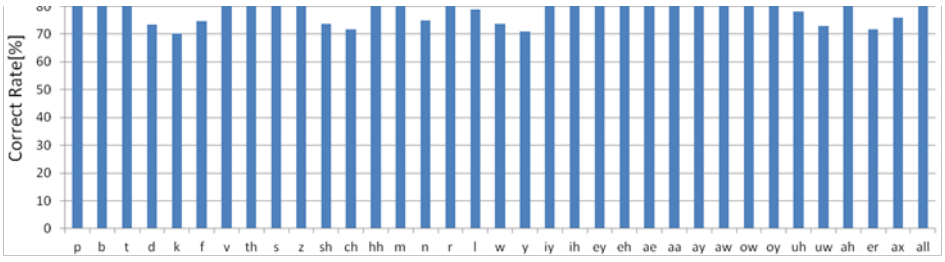


Figure 7: Correct rate of articulatory features

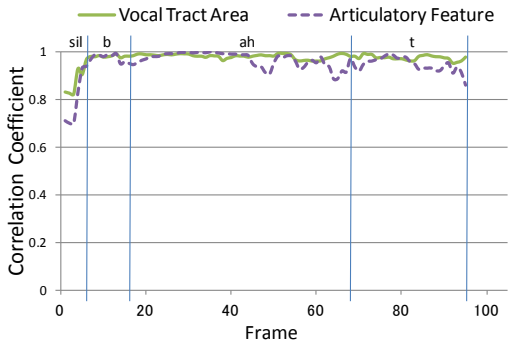


Figure 8: Correlation coefficient of an utterance “but”.

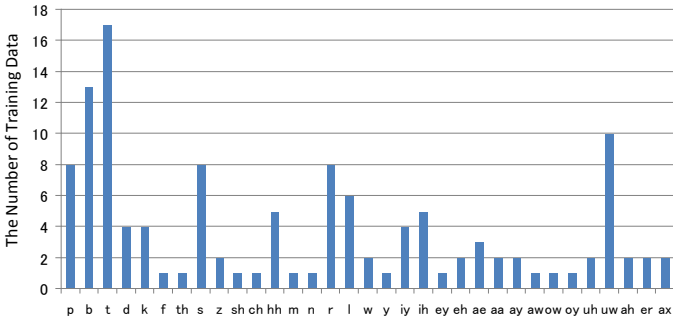


Figure 9: The number of training data for each phoneme.

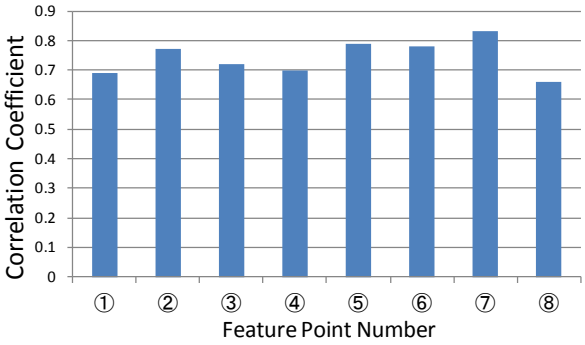


Figure 10: Correlation coefficient of each feature point.

Figure 8 shows an example of the time sequences of the correlation coefficients of an utterance "but". In the figure, we can observe the articulatory movement with high correlation at the preceding part of the utterance “b” (“sil”; from the frame 0 to 7). It seems that animated pronunciation gestures might simulate well even at the rising parts of utterance. Figure 9 shows the number of phonemes contained in the training data. Though the phoneme “t” has comparatively enough data, its correlation coefficient in Figure 6 is not so high. Because there must be a key point in each articulatory gesture, an animated pronunciation software should have such facilities that can focus on the key articulation for each phoneme.

Figure 10 shows the correlation coefficient for each articulatory organ. The horizontal axis shows feature points, that are; from feature point ① to ⑤ refers to the tongue, feature point ⑥ refers to the upper lip, ⑦ refers to the soft palate, and ⑧ the lower lip. Though the soft palate ⑦ shows high correlation, the lower lip ⑧ is not so high degree. Moreover, the averaged correlation coefficient of the important organ, tongue (① to ⑤), is 0.7, and further improvement in detecting articulatory gestures related to tongue positions and lower lip. We have a plan to improve MLNs by focusing to the training of important articulatory manners and positions.

IV. CONCLUSION

We developed a system that can generate CG animation of pronunciation movement by extracting articulatory features from speech. Pronunciation errors of a learner can be seen by displaying the pronunciation movements of his/her tongue, palate, lip, and lower jaw on a screen as a comparative animation with a teacher. Experimental results show that the correlation coefficient of the CG animations with articulatory gestures investigated in the MRI data is 0.83 on the average, and we confirmed that smooth motions in the animations can be generated from speech. Future works include the development of a pronunciation instructor system with CG animation and evaluations of our proposed system by conducting in language classes. In this evaluation stage, we will investigate the possibility of pronunciation error detection for non-native speakers.

REFERENCES

[1] R. Delmonte, “SLIM prosodic automatic tools for self-learning instruction,” *Speech Communication*, 30(2-3):145–166, 2000.

[2] J. Gamper and J. Knapp, “A Review of Intelligent CALL Systems,” *Computer Assisted Language Learning*, 15(4): 329–342, 2002.

[3] S. M. Witt and S. J. Young, “Phone-level pronunciation scoring and assessment for interactive language learning,” *Speech Communication*, *30*(2-3), 95–108, 1995.

[4] O. Deroo, C. Ris, S. Gielen and J. Vanparys, “Automatic detection of mispronounced phonemes for language learning tools,” *Proceedings of ICSLP-2000*, vol. 1, 681–684, 2000.

[5] S. Wang, M. Higgins, and Y. Shima, “Training English pronunciation for Japanese learners of English online,” *The JALT Call Journal*, *1*(1), 39–47, 2005.

[6] Phonetics Flash Animation Project: http://www.uiowa.edu/~acadtech/phonetics/

[7] K. H. Wong, W. K. Lo and H. Meng, ”Allophonic variations in visual speech synthesis for corrective feedback in CAPT,” *Proc. ICASSP 2011*, pp. 5708-5711, 2011.

[8] Y. Iribe, S. Manosavanh, K . Katsurada, R. Hayashi, C. Zhu and T. Nitta, ”Generation animated pronunciation from speech through articulatory feature extraction,” *Proc. of Interspeecch’11*, pp.1617-1621, 2011.

[9] Y. Iribe, S. Manosavanh, K . Katsurada, R. Hayashi, C. Zhu and T. Nitta, ”Improvement of animated articulatory gesture extracted from speech for pronunciation training,” *Proc. of ICASSP’12*, 2012

[10] H. Takemoto, K. Honda, S. Masaki,Y. Shimada, and I. Fujimoto, “Measurement of temporal changes in vocal tract area function from 3D cine-MRI data,” J. Acoust. Soc. Am, 119 (2), pp.1037-1049, 2006.

[11] T. Nitta, T. Onoda, K. Katsurada, ” One-model speech recognition and synthesis based on articulatory movement HMMs,” Proc. Interspeech 2010, pp.2970-2973, 20.

Pronunciation analysis by acoustic-to-articulatory feature inversion

Olov Engwall
Centre for Speech Technology, CSC, KTH, Stockholm, Sweden
engwall@kth.se

***Abstract*—Second language learners may require assistance correcting their articulation of unfamiliar phonemes in order to reach the target pronunciation. If, e.g., a talking head is to provide the learner with feedback on how to change the articulation, a required first step is to be able to analyze the learner's articulation. This paper describes how a specialized restricted acoustic-to-articulatory inversion procedure may be used for this analysis. The inversion is trained on simultaneously recorded acoustic-articulatory data of one native speaker of Swedish, and four different experiments investigate how it performs for the original speaker, using acoustic input; for the original speaker, using acoustic input and visual information; for four other speakers; and for correct and mispronounced phones uttered by two non-native speakers.**

I. Introduction

In recent years, several different research teams have investigated the use of animated talking heads displaying additional information on the intra-oral articulation, to either support speech perception or to practice pronunciation [1]–[5].

The underlying idea is that second language learners, hearing-impaired persons and children with speech disorders may have difficulties aquiring the correct articulation of unfamiliar phonemes and that a display showing and describing tongue positions and movements could contribute to their learning of the correct pronunciation. Instead of showing a normal face view, parts of the skin of the talking head is removed or made transparent, in order to make the intra-oral articulation visible (c.f. [6] for an overview of different alternatives for the display), as examplified in Fig. 1.

However, both perception and production studies using such displays indicate that it is initially difficult for the learners to extract information from animations of tongue movements.

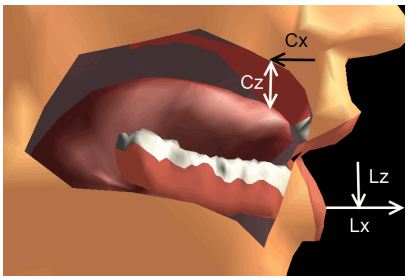


Fig. 1: *Talking head display showing tongue articulations by making parts of the skin transparent. The figure further illustrates the articulatory feature measures defined in Sec. II.*

The pronunciation training studies with quantitative evaluation [3], [4] did not find any significant additional improvement for leaners who were shown animations of tongue movements to imitate, compared to control groups that did not. It has therefore been suggested [5], [6] that it is essential that the visualizations provide feedback related to the learner's *own* articulation, rather than general advice that is unrelated to the learner's attempt. Audiovisual feedback that describes the required articulatory change relative to the learner's tongue position (e.g., "Try to raise the front part of your tongue slightly" accompanied by an animation illustrating the movement and the part of the tongue involved, for the change from [eː] to [iː]) may be more effective than displaying the target tongue position and shape. We have previously found [6] some short time improvement in the learner's articulation (and hence pronunciation) when a virtual teacher instructed French speakers how to change their articulation to produce unfamiliar Swedish phonemes. A pre-requisite for such feedback is that the learner's articulation can be estimated, in order to generate instructions on how to change it. This is the topic of the present paper: acoustic-to-articulatory inversion in the setting of pronunciation training.

Acoustic-to-articulatory inversion normally signifies that one uses the produced speech sound to estimate either the vocal tract shape or specific marker positions (most commonly Electromagnetic Articulography – EMA – coils, placed on relevant articulators). Acoustic-to-articulatory inversion is problematic, since several different combinations of articulator positions may produce the same speech sound and there are too few parameters in the speech signal to fully estimate the vocal tract configuration. In this paper, two different paths are used to alleviate these problems. The first, already suggested in [6], is to restrict the sought articulatory information to only those features that are important for pronunciation training. The second is to add visual information from the speaker's face to obtain more input features to the estimation [7], [8].

The simplification of the inversion problem is based on how pronunciation training is carried out with the virtual teacher: The expected correct input from the learner is known (because the practise is based on e.g., repetitions of teacher utterances, reading a text out loud, or eliciation of specific utterances), and it can therefore be automatically segmented and transfered to an automatic pronunciation analyzer that uses hypotheses about probable errors to judge if the learner's production was correct or not. The hypotheses are founded

on linguistic knowledge about frequently occurring errors for the particular target phoneme, taking into account the learner's first language.

This paper is divided into three different sub-topics with separate experiments and results, and a common discussion of the findings and possible future work. The first part summarizes the proposition made in [6] to define a restricted articulatory inversion (Sec. II), and extends the experiments of that article by investigating how added visual information of lip movements contribute to the estimation of the underlying articulation (Sec. III). The second part (Sec. IV) addresses the problem of speaker dependence of articulatory inversion, by investigating the estimation of articulations for four speakers that the inversion algorithm had not previously been trained on. The third part (Sec. V) is a preliminary investigation of to what extent a general articulatory inversion, trained on a native speaker can separate correctly and incorrectly uttered phonemes by non-native speakers in the articulatory space, using only acoustic input.

II. Acoustic-to-Articulatory feature inversion

As an alternative to pronunciation error detection/analysis based on standard automatic speech recognition, we have proposed [6] to use acoustic-to-articulatory inversion with restrictions to estimate the required information on the learner's articulation. The output from such an inversion is to some extent similar to feature detection [9], [10], which is focused on finding articulatory information (of e.g., place and manner of articulation) in the acoustic signal. However, whereas feature detection results in separation into distinct classes ('front', 'central' or 'back' articulation), articulatory inversion can potentially provide information on articulator position within the classes. For pronunciation training with a virtual teacher these two aspects may be important to adjust the feedback instruction depending on how large the learner's articulatory change needs to be, and on improvements in the articulation compared to the previous attempts (an articulatory change in the correct direction should be acknowledged by the teacher, even if the correct place of articulation was not yet reached).

The method used here relies on training Gaussian Mixture Models (GMM), using simultaneously collected acoustic and articulatory data. It differs from standard articulatory inversion by three main simplifications:

The articulatory information is restricted to the position of the most constricted part in the oral cavity and the vertical distance between the tongue and the palate at this point, rather than the entire vocal tract shape.

The inversion is performed using a separate articulation analyzer for each target phoneme, with the training material consisting of correct target phones plus phones illustrating frequently occurring pronunciation errors for that target, rather than one general inversion trained on the entire corpus.

The targeted pronunciation training only focuses on static articulations, and the analyzer therefore only uses the stable part of the acoustic signal for training and analysis, rather than the entire recording.

A. Data and experimental set-up

The articulatory feature inversion is currently trained on simultaneously recorded acoustic and articulation data from one female speaker of Swedish, but future use of the method would require a database of several native and non-native speakers, in order to create a more adequate mapping between the acoustics of correct and incorrect phones on the one hand and their corresponding articulation on the other. However, for demonstration of the method, the available data can be used with the approximation that the mispronunciation of one phoneme makes it similar to another in the same language.

The database used consists of simultaneous acoustic, video, optical motion capture, EMA and acoustic recordings of 135 symmetric VCV words and 180 simple Swedish noun-verb-object sentences [8], [11]. The current experiments use data from two EMA coils glued midsagittally on the tongue, the first close to the tip and the second 30 mm further back, and four optical motion capture markers, placed at the left and right mouth corners and the upper and lower lips. The data of EMA coil positions, captured with the Movetracksystem [12], was down-sampled to 60 Hz to correspond to the frame-rate of the MacReflex optical motion capture system, resulting in 23,409 frames of data.

Two measures were defined to describe the articulation of the tongue [6]:

C_z the minimal vertical linguopalatal distance.
C_x the horizontal distance between the upper incisor and the point at the palate for which the minimal distance C_z occurs.

In addition, two measures describe the articulation of the lips:

L_x the summed protrusion of the mouth corners.
L_z the summed vertical position of the lips.

The underlying idea is that C_x and C_z describe the place (front-back) and manner (open-close) of articulation efficiently, that L_x captures errors in lip rounding and L_z the difference between bilabials and labiodentals. The measures are illustrated in Fig. 1 and their derivation is described in detail in [6]. This paper concentrates on estimation of the tongue features ($\tau = C_x, C_z$) and instead use the motion capture data for the lips as input to the audiovisual-to-articulatory inversion.

The acoustic signal was a) divided into frames (length 24 ms, shift of 16.67 ms) to correspond to the articulation data frame rate of 60 Hz, b) pre-emphasized and multiplied by a Hamming window, and c) transformed into 16 line spectrum pairs (LSP), using a covariance-based LPC algorithm [13]. This resulted in acoustic data a consisting of 23,409 frames with 16 LSP coefficients and the RMS amplitude.

Each frame was assigned a phonetic label, using an HMM-based automatic aligner [14], and all frames labeled as belonging to a phoneme p were grouped as (a^p, τ^p), after removing initial and final transitions in each sequence

A Gaussian Mixture Regression (GMR) [15] is used to estimate the articulatory measures C_x, C_Z from the acoustic data, employing an Expectation-Maximization (EM) algorithm and k-means clustering initialization. A mapping function is

first defined between the articulatory features $\tau^{training}$ and acoustic data $a^{training}$, by training a set of GMMs on the joint probability density function of the articulatory-acoustic training material. This mapping function is then used to estimate the corresponding articulatory features τ^{test} for given acoustic data a^{test}, for a test material not included in the training.

In the following sections, four different sets of training and test material were used for each analyzed phoneme T:

1) The training material was 9/10th of the acoustic-articulation data for the original speaker S_O for the target phoneme T and another phoneme M that constitutes a frequently occurring mispronunciation of T. The test material was the remaining 1/10th of the data. Training and test parts were rotated in a jack-knife procedure, so that all parts were used for training and test.
2) The training and test material was the same as above, with the exception that the data for the lips ($MC_{x,y,z}$) was used as input to the regression, in order to investigate audiovisual-to-articulatory inversion for the tongue measures. The experiments using these two sets are described in Sec III.
3) The training material was all the data (10/10th) for the original speaker S_O for phonemes T and M, as described above. The input test material was acoustic data for T and M for four other speakers (as described further in Sec IV), in order to investigate speaker independent inversion that used a regression model trained on S_O.
4) The training material was again all the data for the original speaker S_O for phonemes T and M. The test material was correctly and incorrectly produced acoustic data for T and M for two non-native speakers, again using GMR trained on S_O. These experiments are further described in Sec. V.

This paper concentrates on vowel targets, while [6] in addition addresses a number of problematic phonemes. The evaluation corpus consisted of ten vowel pairs (T and M) that are frequently confused by non-native speakers of Swedish: [ɑː-ɑ, eː-ɛː, eː-iː, ʉː-yː, ʊ-ʏ, øː-oː, øː-ɛː, ø-ɛ, ø-ʊ].

III. Audiovisual-to-articulatory inversion

This section compares the inversion results obtained without (i.e., using the training and test set described under 1. above) and with visual information (i.e., using set 2 above). For the audiovisual inversion, late fusion was employed, signifying that acoustic and visual data was first used separately to make two estimations of the articulatory features of the tongue, and that these were then combined into the final estimation, using a weighted sum. The weights for the fusion are defined by the correlation between the true data and the respective reconstructions from acoustic and visual information, as described in more detail in [8].

Three different measures were used to evaluate the inversion results, Pearson's correlation coefficients (CC) and the root mean squared error (RMSE) between the estimated and true articulatory features at each time frame; and the percentage of correct classifications, i.e., if the estimated articulations mapped onto the articulation cluster of the correct phoneme.

The results, summarized in Fig. 2, show that adding visual information leads to a 13% mean increase of the correlation coefficients (from 65% to 78%) for the (C_x, C_y) estimation, a mean reduction of the RMSE of 0.3 mm (from 2.84 to 2.54 mm) for C_x (no change for C_z), and a slight mean increase of 1.6% for correct classifications (from 96.4 to 98.0%). These findings are in line with previous studies comparing general acoustic only and audiovisual articulatory inversion [7], [8] in that the correlation between facial and tongue movements can be used to improve the estimation of the latter. The change is not dramatic, but could be important to avoid erroneous feedback to the learner in pronunciation training.

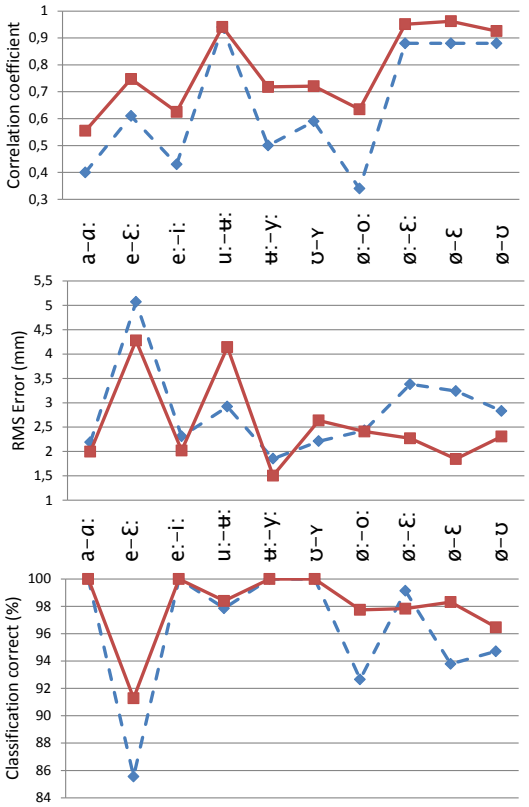
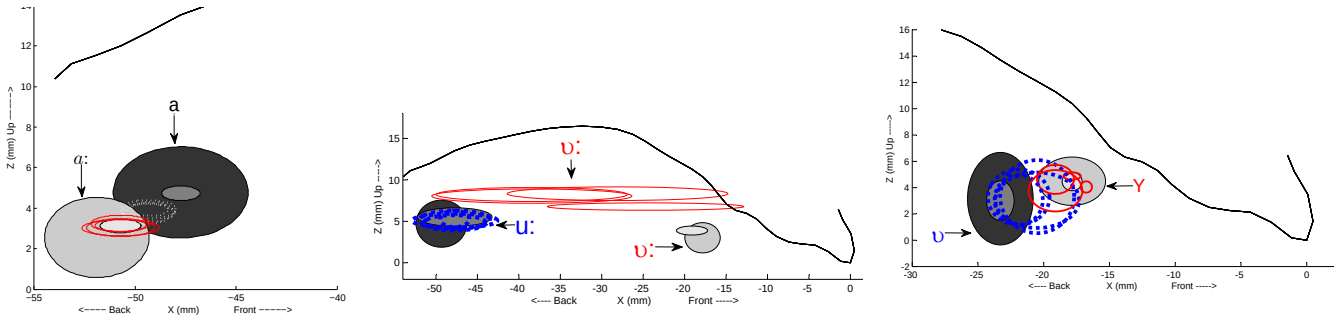


Fig. 2: *Results of the articulatory feature inversion for different pairs of confusable vowels, when based on acoustic only data (red, dashed line) and supported by visual information of lip movements (blue, solid line).*

IV. Speaker independence

This section investigates if GMR trained on one, female, speaker can be used to estimate the articulation of four other, male, speakers (i.e., using data set 3). The data for the four other speakers consisted of acoustic recordings of 118 Swedish one- or two-syllable words constituting minimal pairs only differing in the middle vowel (e.g. *"rita"* - *"ryta"* - *"ruta"* - *"röta"*, draw - roar - window pane - rot) . The corpus was



(a) [aː-ɑ] (b) [uː-ʉ], (υː=[ʉ]) (c) [ʊ-ʏ] (υ=[ʊ])

Fig. 3: *Estimated point of articulation for four speakers (empty ellipses), compared to the actual and estimated point of articulation for the original speaker. The ellipses are centered on the mean of the articulatory features, with axes corresponding to the standard deviation in the x- and z-directions. The filled ellipses describe the actual data and the estimation for the original speaker. For each vowel, the actual data is represented by the darker filled ellipse and the estimation by the lighter. For the other speakers, the first vowel is plotted in dotted blue (white for [a]) and the second in solid red.*

chosen to include all 18 Swedish long and short vowels in different contexts, and was collected to be used as a basis for experiments on vowel mispronunciation detection [16]. Two of the speakers were native and two were non-native at different levels of proficiency and native language (one French speaker, without any previous training in Swedish and one Indian speaker, at an intermediate level of Swedish). For the non-native speakers, two recordings were made. In the first, the speaker read the word presented on the screen, while in the second, the word was in addition first spoken by a native speaker before the non-native speaker's attempt. The utterances of the two non-native speakers were labeled as correct or incorrect by a phonetically trained labeller, and only vowels labeled as correct in the second setting are used for the experiments in this section.

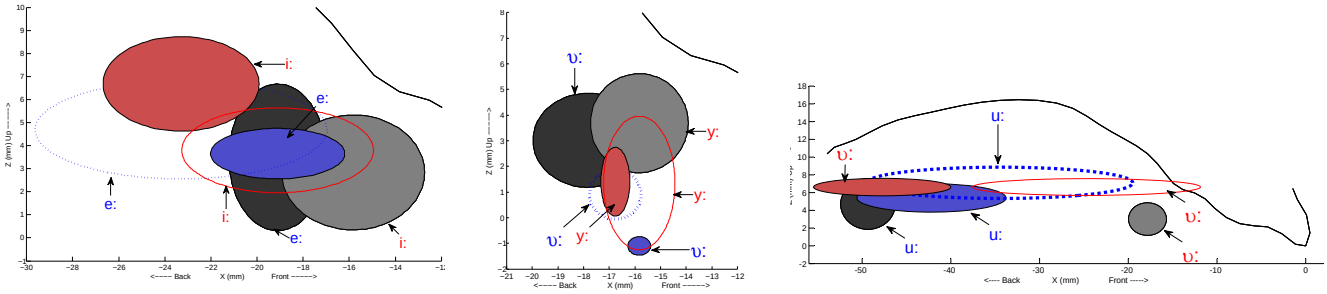
A simple speaker normalization procedure was used, calibrating the LSP values for each of the four speakers by a scaling factor that set the mean value of each LSP to that of the original speaker. A more sophisticated normalization, e.g. Vocal Tract Length Normalization would be more appropriate in future versions of the analyzer, but this simple normalization is adequate for the current experiments.

Since no articulatory data is available for the four speakers, it is not possible to evaluate the inversion results in terms of correspondence with the actual data. Instead, we focus on the extent to which their acoustic signal maps onto the expected articulatory cluster, using four different measures for each speaker: the share of "correct" classifications (*corr*), of estimated place of articulation (C_x, C_z) coinciding with the actual place of articulation for the original speaker (*CH*, correct hit), of intra-speaker overlap between the estimated articulation ellipses for T and M (FH_σ, first type of false hit), and of the overlap between the estimated articulation ellipse of T with that for M for the original speaker (FH_μ, second type of false hit). A successful inversion should in principle lead to high values of *corr* and *CH* and low values of FA_μ and FH_σ. It should however be noted that "correct" is ambiguous in this case, since it is possible that inter-speaker articulation differences may in reality lead to overlap between the place of articulation for phoneme T for speaker 1 and phoneme M for speaker 2. In addition, as illustrated in Fig. 3a), there is in some cases also an articulatory overlap in the actual data for two different phonemes for the same speaker.

Fig. 3 illustrates three different cases in the inversion results. In Fig. 3a), the estimation for all four speakers for both phonemes in the contrasting pair maps onto the corresponding areas of articulation for the original speaker ($CH = 1$), and there is no overlap between the two tested phonemes for any of the speakers (FH_σ=0). The overlap between the estimations and the other articulation areas for the original speaker (FH_μ) is also present in the actual data. In Fig. 3b), the estimation maps on the expected cluster for all four speakers for one of the phonemes ([uː]), but not for the other ([ʉː]) ($CH = 0.5$). There is however no overlap between the estimations for [ʉː] and the articulation areas of [uː] for the original speaker (FH_μ=0) or the estimations for the other speakers (FH_σ=0). In Fig. 3c), the estimations of [ʊ] and [ʏ] roughly maps to the corresponding articulation areas for the original speaker, but for two speakers there is an overlap with the articulation area of [ʏ] for the original speaker (FH_μ=0.25), and for one there

TABLE I: Measures for evaluation. Share of correct classification (*corr*), of mapping onto the correct articulation of the original speaker(*CH*), of intra-speaker overlap between the two vowel articulations (FH_μ), and of inter-speaker overlap with the articulation of the other vowel for the original speaker (FH_σ). $\widehat{Sp_O}$ is the estimate for the original speaker Sp_O

	Sp_O	$\widehat{Sp_O}$	$\widehat{Sp_1}$	$\widehat{Sp_2}$	$\widehat{Sp_3}$	$\widehat{Sp_4}$
corr	0.87	0.91	0.66	0.68	0.70	0.57
CH	-	0.90	0.80	0.75	0.90	0.75
FH_σ	0.40	0.30	0.20	0.30	0.20	0.30
FH_μ	-	0.30	0.20	0.30	0.20	0.40



(a) [eː-iː] (b) [yː-ʉː] (c) [oː-ʉː]

Fig. 4: *Estimated point of articulation (mean and standard deviation) for phones labeled as correctly pronounced (empty ellipses) and mispronounced (filled coloured ellipses) for two non-native speakers, compared to the actual point of articulation for the original speaker (grey-scale filled ellipses). For each pair, the first vowel is plotted in dark grey and dotted blue, and the second in light grey and solid red.*

is overlap between the estimations of [ʊ] and [ʏ] (FH_σ=0.25). The quantitative results are summarized in Table I. For the correct classification measure, the classification score for the estimated positions $\widehat{Sp_O}$ is higher than for the original data, since the regression makes an estimation closer to the mean and may have less standard deviation than the original data (this is also illustrated by the size of the ellipses in Fig. 3). The correct classification is 20-30% lower for the speakers not used for training than for the original speaker, but for three of them, the separation between the two vowels is nevertheless rather successful, in particular considering 1) that the decrease might partly be due to actual articulation differences between the speakers and 2) that the differences in tongue position between some vowel pairs are small (such as between [ɑː] and [a] or [ʊ] and [ʏ]). The result for the non-native beginner level speaker, Sp_4, is substantially worse, with lower *corr* and larger overlap between T and M (FH_σ=0.4), indicating that his pronunciation differed from the other four speakers and that this is reflected in the articulation estimations.

Considering the vowels, the articulation ellipses for [eː, iː, yː, oː, uː, aː, ɔ, ʊ, ʏ, ɑ] were always estimated as coinciding with those of the original speaker ($CH = 1$) for all speakers and all pairs; [ɛː, øː, ʉː] were slightly more problematic, with $0.25 < CH < 1$, depending on the vowel pair; and [ø] was mostly estimated at a different place of articulation than the original speaker ($0 < CH < 0.5$). For the vowel pairs, FH_σ overlap between the T and M articulation ellipses occured for [ø-ʊ, ø-ɔ] (for 3 speakers) and [eː-ɛː, øː-ɛː] (for 2 speakers).

From the above results, it seems that Gaussian Mixture Regression trained on one speaker can, to large extent, be used to discriminate articulatorily between two similar vowels produced by other speakers, at least for vowel pairs not involving the more central and mid-open vowels [ɛː, øː, ø]. This is positive for the potential use in a pronunciation training system employing articulatory feedback, but at least three words of caution need to be added. The first is that we are here dealing with the mean and the standard deviation of the articulation estimate over several occurrences produced in a corpus, whereas an articulatory feedback system needs to analyze a single occurrence of T, and it remains to be shown that this latter estimate is similar to the mean for all occurrences of T. The second is that T and M are here two correctly produced different phonemes, whereas the articulatory feedback system would deal with T and mispronunciations of T. The analysis principle is the same, and since T and M were selected here to represent frequently confused phonemes, M may in fact be a good approximation of the mispronunciations of T, but it is plausible that mispronunciations of T may differ from both T and M. The third is that the articulation of the same phoneme may differ between speakers, and an articulatory estimation based on training data from one speaker might therefore result in erroneous feedback, even if the analysis of the (acoustic) pronunciation is correct (i.e., the regression identifies a point of articulation (C_x, C_z) that would have been correct if the phone was uttered by the original speaker, but the change that the original speaker would have to make from (C_x, C_z) to the target might not be the same as the change that the speaker who actually uttered the phone needs to make). This last issue can only be investigated and resolved through larger acoustic-articulatory databases of several native and non-native speakers, to evaluate and train the correspondence between the estimated point of articulation and the actual data. The two first issues are tentatively addressed through a preliminary and small study in the following section.

V. Feature Inversion of Mispronounciations

This section briefly looks into the problem of comparing the estimated articulation for correct and incorrect non-native utterances of a phoneme. In order to do so, the labeled data of the two non-native speakers was used, grouping for each phoneme utterances that were labeled as correct on the one hand and those labeled as incorrect on the other. The two sets of data were used separately as input to the gaussian mixture regression to provide an estimate of the articulation in the two cases. Since the amount of data was very small, the two speakers' data were concatenated, after speaker normalization.

For several of the vowel pairs, there was nevertheless too little correct or incorrect data for the training, and they had to be excluded from the analysis below. Only the pairs [eː-iː, ʉː-yː, oː-ʉː, øː-oː, ø-ɔ, ɑː-ɑ] included enough data of both correct and mispronounced phones. Fig. 4 shows the result for the first three of these pairs, illustrating that for several phonemes ([eː, iː, ʉ, yː]), there was indeed a difference in the estimated articulation between the phones labeled as being correct and those labeled as incorrect, and the misprounced phones were estimated articulatorily closer to the other vowel in the pair. Fig. 4c) however shows two of the remaining problems. First, that even if correct and mispronounced phones of [ʉː] are following the hypothesis of being closer to the original speaker's articulation of [ʉː] and [uː], respectively, the same is not true for the correct and mispronounced phones for [uː]. Second, that the articulatory estimation will, quite naturally, depend heavily on the training material: the point of articulation for [ʉː] is estimated differently in Fig. 4b)–c), because the other vowel in the pair differs. This shows the importance of linguistic pre-processing and/or a tailored training material for the analyzer.

Linguistic pre-processing signifies that a hypothesis on the expected pronunciation errors is formed for each tested phoneme, based on the speaker's native language and the context the phoneme appears in, and that the correct feature analyzer is used to capture this potential error (e.g., with [ʉː] being the target, the mispronunciation trained on the distinction in the pair [ʉː-yː] could be used for learners from Germany and France, whereas [ʉː-oː] would be more appropriate for learners from England and Greece).

Tailoring the training material signifies, as already mentioned in the previous section, that a future version of each analyzer should be trained on correct and mispronounced phones for the same phoneme, rather than correct phones from different phonemes. Using a larger such training material might also resolve the problem observed for [øː-oː, ø-ɔ, ɑː-ɑ], which had partial overlap between the articulatory estimations for correct and mispronounced phones.

The experiments and results presented in this section are very limited, and not in total agreement, and they do by no means constitute any binding proof of the possibility to correctly estimate the difference in articulation between correct and incorrect utterances. The fact that a regression technique trained on one single native speaker can in several cases articulatorily separate mispronounced phones from correct ones for non-native speakers is nevertheless an important indication that the presented analysis method is worth future efforts to refine it.

VI. Discussion & Future directions

The four experiments above do show, on the one hand, that there is potential for the method as an articulation analyzer. They also show, however, that the method is far from error-free and much more work remains before it could be used for actual pronunciation analysis in a real-time computer-assisted language learning system. A first necessary step is to improve and investigate the performance of the method using a larger acoustic-articulatory database of native and non-native speakers. Such a database is however yet to be collected. In wait of this database, tests with an available larger set of acoustic-only data of non-native speaker utterances [17] would also be beneficial to expand the experiments to more speakers and more examples of mispronounced data.

Another direction of future work is to continue investigating how the output from an articulatory analyzer can be conveyed to a learner in an efficient manner to help her correct her articulation. This is non-trivial, in particular for the minor changes that are required to change the articulation from one close vowel to another (c.f., e.g., Fig. 4a).

VII. Acknowledgments

This work is supported by the Swedish Research Council project 80449001 Computer-Animated LAnguage TEAchers (CALATEA).

References

[1] Engwall, O. and Bälter, O., "Pronunciation feedback from real and virtual language teachers," Computer Assisted Language Learning, 20(3):235–262, 2007.

[2] Fagel, S. and Madany, K., "A 3-D virtual head as a tool for speech therapy for children," in Proceedings of Interspeech, 2643–2646, 2008.

[3] Massaro, D. and Light, J., "Read my tongue movements: Bimodal learning to perceive and produce non-native speech /r/ and /l/," in Proceedings of Eurospeech, 2249–2252, 2003.

[4] Massaro, D., Bigler, S., Chen, T., Perlman, M., and Ouni, S., "Pronunciation training: The role of eye and ear," in Proceedings of Interspeech, 2623–2626, 2008.

[5] Ben Youssef, A., Hueber, T., Badin, P., and Bailly, G., "Toward a multi-speaker visual articulatory feedback system," in Proceedings of Interspeech, 2011.

[6] Engwall, O., "Analysis of and feedback on phonetic features in pronunciation training with a virtual teacher," Computer Assisted Language Learning, 25:37–64, 2012.

[7] Katsamanis, A., Papandreou, G., and Maragos, P., "Face active appearance modeling and speech acoustic information to recover articulation," IEEE Transactions on Audio, Speech and Language Processing, 17(3):411–422, 2009.

[8] Kjellström, H. and Engwall, O., "Audiovisual-to-articulatory inversion," Speech Commun., 51(3):195–209, jan 2009.

[9] Frankel, J., Wester, M., and King, S., "Articulatory feature recognition using dynamic bayesian networks," Computer, Speech and Language, 21(4):620–620, 2007.

[10] Teppermann, J. and Narayanan, S., "Using articulatory representations to detect segmental errors in nonnative pronunciation," IEEE Transactions on Audio, Speech and Language Processing, 16(1):8–22, 2008.

[11] Beskow, J., Engwall, O., and Granström, B., "Resynthesis of facial and intraoral motion from simultaneous measurements," in Proceedings of International Congress of Phonetical Sciences, 431–434, 2003.

[12] Branderud, P., "Movetrack – a movement tracking system," in the French-Swedish Symposium on Speech, 113–122, 1985.

[13] Sugamura, N. and Itakura, F., "Speech analysis and synthesis methods developed at ECL in NTT," Speech Commun., 5:199–215, 1986.

[14] Sjölander, K., "An HMM-based system for automatic segmentation and alignment of speech," in Proceedings of Fonetik, 93–96, 2003.

[15] Calinon, S., Guenter, F., and Billard, A., "On learning, representing and generalizing a task in a humanoid robot," IEEE Transactions on Systems, Man and Cybernetics, Part B, 37(2):286–298, 2007.

[16] Picard, S., Ananthakrishnan, G., Wik, P., Engwall, O., and Abdou, S., "Detection of specific mispronunciations using audiovisual features," in Proceedings of AVSP, 2010.

[17] Koniaris, C. and Engwall, O., "Phoneme level non-native pronunciation analysis by an auditory model-based native assessment scheme.," in Proceedings of Interspeech, 2011.

Enhancing the Confidence Measure for An Arabic Pronunciation Verification System

Sherif Abdou[1,3], Mohsen Rashwan[2.3]
1: Faculty of Computers at Cairo University Egypt
2: Faculty of Engineering at Cairo University Egypt
3: The Research and Development International (RDI) Egypt
(sheriff.abdou , mrashwan)@rdi-eg.com

Hassanin Al-Barhamtoshy, Kamal Jambi, Wajdi Al-Judaibi
Faculty of Computing & Information Technology
King Abdulaziz University, Saudi Arabia
(hassanin , kjambi, waljedaibi)@kau.edu.sa

Abstract—**Articulation features are more natural representatives for the speech signal. In this paper we introduce an effort to utilize articulation features for confidence scoring in a Computer Aided Pronunciation Learning (CAPL) system. An HMM model is trained to classify the user utterance on binary manner features. An articulation based confidence is estimated based on the matching degree between the classified manner features and the reference manner hypothesis. The proposed confidence was evaluated with a test data set of 1 hour that was labeled and segmented manually and achieved an average accuracy of 92.6% for matching with expert judgment. Also when integrated in the CAPL HAFSS system the proposed articulation based confidence measure managed to reduce the FALSE rejections by 25% but with minor degradation on the errors detection rate.**

Keywords- Computer Aided Pronunciation Learning; confidence measuring; articulation features

I. INTRODUCTION

Computer Aided Pronunciation Learning (CAPL) has considered attention in recent years. Therefore, many research efforts have been done to improve such systems especially in the field of second language teaching [1][2].

An important component in such systems is the confidence score that is estimated to measure the goodness of the produced target speech. Based on that score the system takes a decision of either accepting the user input as correct or reporting a detected error. Several methods have been proposed for estimating confidence scores that included HMM-based scores, segment classification scores, segment duration scores and timing scores [3]. Later on the HMM based scores was improved by deploying posterior probabilities of phone segments instead of log-likelihoods as it was found to correlate better with human raters than both the duration based scores and the log likelihood based scores [4]. The posterior probability approach was extended to the assessment of individual phonemes instead of whole utterance scores by computing phoneme level score based on average posterior probability score of a large number of utterances [5].

The acoustic models based confidence scores approach has an advantage in the implementation; the score can be obtained easily from the ASR system. However, it has disadvantage in the specialization for the specific phonemes with which L2 learners make frequent errors. In the beginning stage, L2 learners tend to make pronunciation errors on L2 phonemes which do not exist in their native language (L1), and some of these errors may remain even after several years of learning. The pronunciation training methods need to take special consideration for these phonemes, but it is difficult for the acoustic based confidence score method since the scores are calculated for all phonemes in a similar way.

In phonology phonetic units can be described with distinctive features. These features are grouped into categories according to the natural classes of segments they describe: manner features, and place features. The place of articulation of a consonant is the point of contact, where an obstruction occurs in the vocal tract between an active (moving) articulator (typically some part of the tongue) and a passive (stationary) articulator (typically some part of the roof of the mouth). The manner of articulation describes how the tongue, lips, and other speech organs are involved in making a sound make contact. For any place of articulation, there may be several manners [6]. Articulation features are more discriminative than the acoustic features [7]. For example the phone /ʔ/ (همزة) (glottal stop) as in /ʔ/a/n/t/ (أنت) and the phone /h/ (هاء) as in /h/a/z/a (هذا) have very close acoustic features and can be classified as the same class by mistake. However, each one belongs to a different articulation feature class. The latter is fricative while the former is plosive.

Inspired by their discriminative power the articulation features were used to develop confidence scores for some CAPL systems such as the one of [8]. In that system a group of Support Vector Machines (SVM) were trained to detect the landmarks of distinctive features. The method was tested for L2 learners and achieved significantly higher F-score than the acoustic based confidence scoring. Also the combination of the two methods achieved further improvements.

One of challenging applications for CAPL is the automatic training for correct recitation of the holy Qur'an for Arabic speakers. In contrast to the foreign language training task, where a wide variety of pronunciations can be accepted by native speakers as being correct, the holy Qur'an has to be recited the same way as in the classical Arabic and the tolerance for allowed variation is very fine. The Tajweed science is a set of rules studied by reciters to properly pronounce Quran. Such rules are typical mappings to linguistics and articulatory phonology. Each Arabic phone has a completely determined manner and place of articulation which varies according to its context among speech. To recite

Quran, one should learn such articulation features to spell it the exact way. This highlights the importance of articulation features in Tajweed, which is much older than the science of phonetics.

II. SYSTEM DESCRIPTION

The proposed system will use a state of art speech recognizer to detect errors in the speech of the users, as illustrated in figure 1. The overall system modules are the HMM training module which is used to train the system from the training data, the decoder which is used for pronunciation verification and the adaption module to tune the system model to match the acoustic characteristics of the system user.

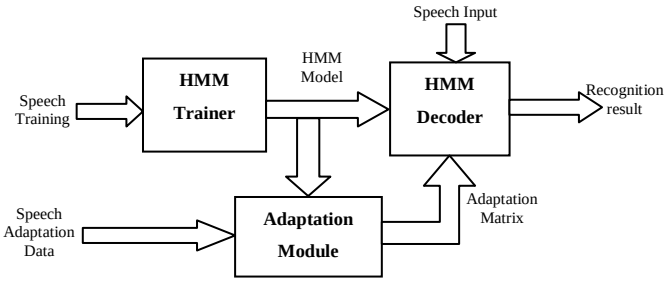


Figure 1. The Oveall System Modules

The HAFSS system is a CAPL system that was developed for the automatic training for Tajweed[9]. The system uses a speech recognizer to detect errors in user recitation. To increase accuracy of the speech recognizer, only probable pronunciation variants, that cover all common types of recitation errors, are examined by the speech decoder. The decision reached by the recognizer is accompanied by a confidence score to reduce effect of misleading system feedbacks to unpredictable speech inputs. The confidence score play a crucial rule in HAFSS system and is used to choose suitable feedback response to the learner. When the system suspects the presence of a pronunciation error with low confidence score the system has some alternate responses:-

- Omit the reporting of the error at all (which is good for novice users because reporting false alarms discourages them to continue learning correct pronunciation).
- Ask the user to repeat the utterance because it was not pronounced clearly.
- Report the existence of an unidentified error and ask the user to repeat the utterance (which is better for more advanced users than ignoring an existent error or reporting wrong type of pronunciation error).
- Report most probable pronunciation error (which if wrong- can be very annoying to many users).

The confidence score of the HAFSS system is based on the Likelihood ratios [10] which can be considered an approximate for posterior probability score.

In this paper we introduce an effort to enhance the confidence score of the HAFSS system using articulation features. In the following sections of this paper, section 2 includes a description of the HAFSS system architecture and the acoustic based confidence score. Section 3 describes the developed articulation based confidence score. Section 4 includes evaluation results for the confidence scores. Section 5 includes conclusions and proposals for future work.

Figure 2 Shows the block diagram of the HAFSS system. Its main blocks are:

- **Verification HMM models**: Used for the acoustic HMM models of the proposed system.
- **Speaker Adaptation**: Used to adapt acoustic models to each user acoustic properties in order to enhance system performance.
- **Pronunciation hypotheses generator**: It analyzes current prompt and generates all possible pronunciation variants that are fed to the speech recognizer in order to test them against the spoken utterance [11].
- **Confidence Score Analysis**: It receives n-best decoded word sequence from the decoder, then analyzes their scores to determine whether to report that result or not.
- **Phoneme duration analysis**: For phonemes that have variable duration according to its location in the Holy Qur'an, this layer determines whether these phonemes have correct lengths or not.
- **Feedback Generator**: Analyze results from the speech recognizer and user selectable options to produce useful feedback messages to the user [10].

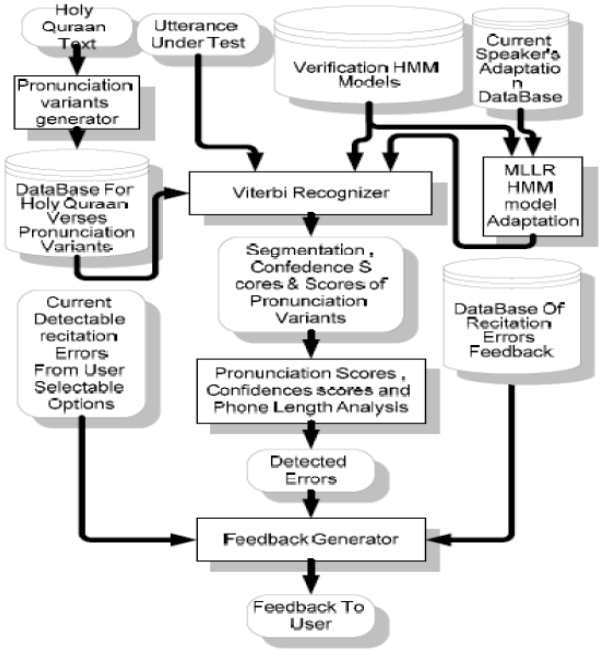


Figure 2. The HAFSS System Block Diagram

The implemented confidence scoring in the system is based on the Likelihood ratios [10] where the acoustic model

likelihoods are scaled by the likelihood of the first alternative path model as the competing decode model. During the decoding process, the Viterbi decoder at the end of each decoded sub-word M_{Best} – at frame x_E - backtracks in the recognition lattice at both the decoded path and the first alternative path M_{1st_alt} until it reaches the node where the two paths meet at the same frame x_S. Then it calculates the average confidence score per frame using the formula:

$$AMC = \frac{1}{N}\sum_{t=S}^{E}\frac{P(x_t \mid M_{best})}{P(x_t \mid M_{1st\_alt})} \quad (1)$$

Where AMC is the Acoustic Model based Confidence measure and N is the number of frames, $N = E - S$. Due to the fact that the difference between these two paths may be significant only in small portion of the path, these small portions should have the most significant effect on the computed confidence score. Therefore, the confidence score of each path is weighted by the distance between the two competing models estimated using Euclidian distance between center of gravity of the two probability distributions [10].

III. Articulation Based Confidence Score

According to Tajweed rules each phone has a specific manner and place features. In this work we focused on manner features. The manner features comes in antonym pairs. Table I shows sample of these features and Table II and III show the used Arabic phone set in the HAFSS system with their assigned manner features.

TABLE I. Sample Tajweed Manner Features

Symbol	Feature name	
	Arabic	Meaning
A	الاطباق	Adhesion
S	الانفتاح	Separation
E	الاستعلاء	Elevation
L	الاستفال	Lowering
F	الاذلاق	Fluency
D	الاصمات	Desisting
P	الشدة	Plosiveness
C	الرخاوة	Fricativeness
V	الجهر	Voicing
U	الهمس	Unvoicing

We trained an GMM model for each manner feature. This model included a number of mixtures in the range 50-300 that was selected us*ing* a development data set. We used the same MFCC features that are used for the base HAFSS models.

The HAFSS system output is evaluated using the articulation models. The articulation confidence measure is evaluated based on the degree of matching of the classified manner feature and the reference phone manner. As shown in Figure 3 the articulation confidence is measured by the ratio between the number of frames that was classified with a manner feature matching the reference phone manner and the total number of frames in that phone based on the output of the HAFSS system hypothesized segmentation.

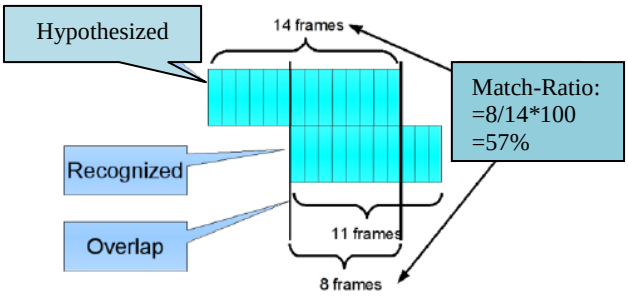


Figure 3. The Calculation of the Articulation Confidence

So for each phone in the proposed output of the HAFSS system the manner matching ratio is estimated by:

$$r_{mp} = \frac{n_o}{n_h} \quad (2)$$

Where r_{mp} is the matching ratio for manner m of phone P and n_o is the number of overlapped frames between the classified manner feature segment and the hypothesized phone segment and n_h is the length of the hypothesized phone segment.

The phone articulation confidence is measured by the average of those matching ratios according to (3).

$$ARC_p = \frac{\sum_{j=1}^{M} r_{mp}}{M} \quad (3)$$

Where ARC_P is the articulation confidence of phone P and M is the total number of evaluated manners.

Our evaluation results, as displayed in the next section, have shown that some manner features are not reliable enough to be considered as confidence predictor. So we decided to discard the manner scores that are less than a specified threshold T. So equation (3) was modified to be:

$$ARC_p = \frac{\sum_{j=1}^{M} \overline{r_{mp}}}{M} \quad (4)$$

$$\overline{r_{mp}} = \begin{cases} r_{mp} & r_{mp} > T \\ 0 & otherwise \end{cases}$$

TABLE II. MANNER FEATURES FOR THE ARABIC PHONES

Phone		Features				
Arabic	Symbol	A/S	E/L	F/D	C/P	V/U
غنة نون مخفاه	n3	S	L	F	C	V
ق	q	S	E	D	P	V
ر مرققة	r	S	L	F	C	V
ر مفخمة	R	S	E	F	C	V
س	s	S	L	D	C	U
ص	S	A	E	D	C	U
ش	s-h	S	L	D	C	U
ت	t	S	L	D	P	U
ط	T	A	E	D	P	V
ث	t-h	S	L	D	C	U
ضمّة	u	S	L	D	C	V
و	w	S	L	D	C	V
نون مدغمة قبل و	w1	S	L	D	C	V
خ	x	S	E	D	C	U
ي	y	S	L	D	C	V
نون مدغمة قبل ي	y1	S	L	D	C	V
ز	z	S	L	D	C	V
ذ	-z	S	L	D	C	V
ظ عامية	Z	A	E	D	C	V
ظ	-Z	A	E	D	C	V

TABLE III. MANNER FEATURES FOR THE ARABIC PHONES (CONT)

Phone		Features				
Arabic	Symbol	A/S	E/L	F/D	C/P	V/U
ع	-@	S	L	D	C	V
أ	@	S	L	D	P	V
فتحة مرققة	a	S	L	D	C	V
فتحة مفخمة	A	S	E	D	C	V
ب	b	S	L	F	P	V
د	d	S	L	D	P	V
ض	D	A	E	D	C	V
ف	f	S	L	F	C	U
ج قهرية	g	S	L	D	P	V
غ	g-h	S	E	D	C	V
ه	h	S	L	D	C	U
ح	-h	S	L	D	C	U
كسرة	i	S	L	D	C	V
ج فصيحة	j	S	L	D	P	V
ج شامية	j-h	S	L	D	C	V
ك	k	S	L	D	P	U
قلقلة	k-1	S	L	D	C	V
ل مرققة	l	S	L	F	C	V
ل مفخمة	L	S	E	F	C	V
م	m	S	L	F	C	V
غنة ميم مشددة	m1	S	L	F	C	V
ميم مخفاة	m3	S	L	F	C	V
مد ا مرقق	m-a	S	L	D	C	V
مد ا مفخم	m-A	S	E	D	C	V
مد ي	m-i	S	L	D	C	V
مد و	m-u	S	L	D	C	V
و مد لين	m-w	S	L	D	C	V
ي مد لين	m-y	S	L	D	C	V
ن	n	S	L	F	C	V
غنة نون مشددة	n1	S	L	F	C	V

IV. EXPERIMENTAL RESULTS

The HMM models were trained using 9 hrs database that contained utterances representing the recitations of randomly selected users of different gender, age and proficiency combinations. These utterances was transcribed by a number of language experts, and labeled with the actual pronounced phonemes. One hour of the training data was segmented manually to determine the exact phone boundaries and the rest of the data was segmented automatically. We used a developing dataset of 16 minutes that is manually segmented and transcribed and consists of 512 utterances from 12 speakers (5 males, 2 females, 5 children). This development set was used for tuning the training HMM parameters and setting the confidence threshold *T*. Finally a test set of 64 minutes that consists of 2230 utterances from 31 speakers (17 males, 8 females, 6 children) manually segmented and transcribed was used for the confidence measure evaluation.

In the first experiment we evaluated the accuracy of the trained manner HMM models by counting the percentage of the evaluation data that the models managed to successfully classify the manner feature. Figure (4) displays the accuracies for the manner classifiers for the different phones. The combined confidence scores are displayed in figure (5). It can be noticed that for most phones the proposed measure achieved high performance. However, some phones still have poor performance and this can be due to not having enough counts in the test data. The average confidence score is 92.58%. For the ASR system, defining a confidence threshold below this value should provide a reliable confidence measure.

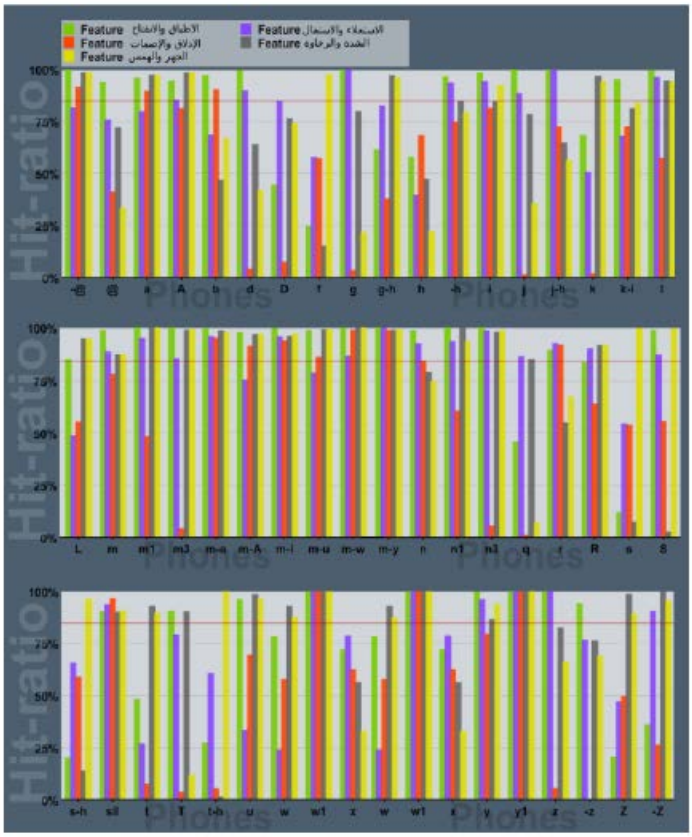


Figure 4. The Manner Features Classifications Accuracies

Figure 5. The Articulating Confidence Measure Accuracy

In the last experiment we evaluated the performance of the articulation confidence against the old confidence measure of the HAFSS system. It was necessary to apply the confidence scores in the same way the old confidence measure was applied.

This can be summarized as follows:

- If the recognized phone matches the corresponding reference phone that the user was requested to recite, then HAFSS accepts the phone as correct and bypasses the confidence score.
- Otherwise, HAFSS uses the confidence score given to that phone as follows:
  - If it is greater than a certain predefined threshold (confidence threshold), then HAFSS is confident about its recognition and reject that phone prompting the user of his/her error.
  - Otherwise, HAFSS is not sure if its recognition was correct (low confidence), so it prompts the user to repeat his/her recitation because the phone was not clear.

The utterances used in this experiment contained a 3.4% of error in the recitation. This was manually identified by human experts. It is required from HAFSS to correctly identify errors and accept most of the correct parts. For the new confidence measure to be robust and useful, repeat requests should be minimized as much as possible for the erroneous parts. Because it is already known that the speakers made an error, it is desired that the system prompts for such errors. As for the correct parts, it is desired to give low confidence for corresponding phones in order not to reject the correct phones, but to prompt the user to repeat such phone as they were not clear. As for wrong parts of speech that are identified as correct (the recognized phones match the reference phones), the confidence scores are not used, so the phones go undetected. The following points summarize various performance measures to be calculated from the system:

- True Acceptance Ratio (TAR): percent of correct speech that HAFSS recognizes as correct.
- True Rejection Ratio (TRR): percent of wrong speech that HAFSS recognizes as wrong.
- False Acceptance Ratio (FAR): percent of wrong speech that HAFSS recognizes as correct.
- False Rejection Ratio (FRR): percent of correct speech that HAFSS recognizes as wrong.
- Wrong Repeat Request Ratio (WRR): percent of wrong speech that HAFSS prompts for repeat request.
- Correct Repeat Request Ratio (CRR): percent of correct speech that HAFSS prompts for repeat request.
- Rejection confidence for wrong speech: ratio of TRR to WRR. It represents how much the system rejection is confident for wrong speech. It is desired to maximize this measure.
- Rejection confidence for correct speech: ratio of FRR to CRR. It represents how much the system rejection is confident for correct speech. It is desired to minimize this measure.

Table IV illustrates the meaning of various performance measures.

TABLE IV. RELATION BETWEEN HUMAN JUDGMENT AND SYSTEM JUDGMENT

System judgement	Human judgement	
	Wrong	Correct
Correct	FAR	TAR
Wrong	TRR	FRR
Repeat Request	WRR	CRR
Rejection Confidence	TRR/WRR	FRR/CRR

These measures are optimized by varying the confidence threshold and recalculating them. However, TAR and FAR remain constant because they do not depend on the threshold where the confidence score is bypassed. The results are illustrated graphically in Figure 6 as stacked vertical bars. The X-axis denotes the threshold value while the Y-axis denotes the percent of speech. TAR was omitted from the graph because it occupies most of the data set and it is constant, so no need to display it.

Table V shows the HFASS system performance using the articulation based confidence measure and threshold value of 92.6% and Table VI shows the HAFSS system performance using its old acoustic based confidence measure and the same test data set.

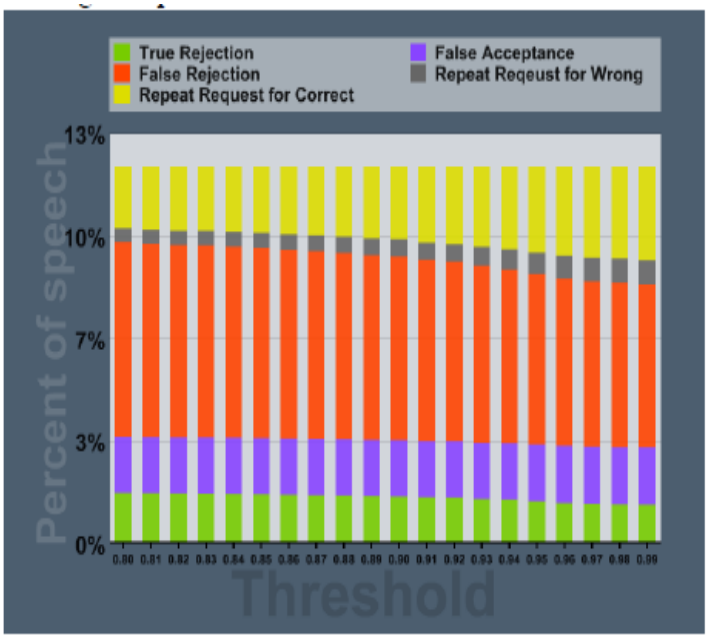


Figure 6. HAFSS Recognition Results with new Confidence Measure

We can see that the articulation confidence measure managed to reduce the FRR rate from 7.8% to 5.8%. This means 25% reduction in the system false rejection responses compared with acoustic based confidence. This improvement was with the price of minor reduction in the detected errors as the TRR decreased from 1.5% to 1.4%.

TABLE V. HFASS SYSTEM PERFORMANCE USING THE ARTICULATION BASED CONFIDENCE

System Judgment	Human Judgment	
	Wrong	Correct
Correct	1.6	88%
Wrong	1.4	5.8
Repeat	0.4	2.9

TABLE VI. HFASS SYSTEM PERFORMANCE USING THE ACOUSTIC BASED CONFIDENCE

System Judgment	Human Judgment	
	Wrong	Correct
Correct	1.6	88%
Wrong	1.5	7.8
Repeat	0.3	1.9

V. CONCLUSIONS

In this work we introduced an articulation based confidence measure that is based on the matching of the recognized manner feature with the hypothesized phone manner feature. The percentage of matched frames was used as a measure for confidence in the judgment of a Computerized Aided Pronunciation Learning HAFSS system. The proposed measure managed to reduce the system FALSE rejections by 25% compared with an acoustic model based confidence measure but with minor degradation on the errors detection rate. In future work we plan to investigate using other types of classifiers for the articulation features such as Support Vector Machines (SVM) that has shown superior performance [7] [8]. Also we will investigate integrating the two types of confidence measures using a weighting approach that favor the measure with the more reliable performance for each phone.

ACKNOWLEDGMENTS

Authors of this paper thankful to KACST through their grant's number 10-INF-1406-03. Their financially and support during the period of this research took place is greatly acknowledged. Also the first author was partially supported by the ADEPT project funded by the Middle East and North Africa - Swedish Research Links Programme (MENA).

REFERENCES

[1] L. Neumeyer, H. Franco, V. Digalakis, and M. Weintraub, "Automatic scoring of pronunciation quality," in Speech Communication, 2000, pp. 88–93.

[2] S. Witt, "Use of the speech recognition in computer-assisted language learning," Unpublished dissertation, Cambridge University Engineering department, Cambridge, U.K., 1999.

[3] H. Franco, L. Neumeyer, Y. Kim, and O. Ronen, "Automatic pronunciation scoring for language instruction". Proc. International Conference on Acoustics, speech, and Signal Processing, pp. 1471-1474, Vol. 2, Munich, Germany, 1997.

[4] H. Bratt, L. Neumeyer, E. Shriberg, and H. Franco, "Collection and Detailed Transcription of a Speech Database for Development of Language Learning Technologies", Proc. Intl. Conf. on Spoken Language Processing, Sydney, Australia, 1998

[5] F. De Wet, C. Cucchiarini, H. Strik, and L. Boves, "Using likelihood ratios to perform utterance verification in automatic pronunciation assessment", Proc. Of Eurospeech-99, Budapest, Hungary, pp. 173-176, 1999.

[6] G. A. Miller and P. E. Nicely. "Analysis of perceptual confusions among some English consonants". Journal of the Acoustical Society of America, 27:338–352, 1955.

[7] M. Hasegawa-Johnson, J. Baker, S. Borys, K. Chen, E.Coogan, S. Greenberg, A. Juneja, K. Kirchhoff, K. Livescu, S. Mohan, J. Muller, K. Sonmez, and T. Wang, "Landmark-based speech recognition: Report of the 2004 johns hopkins summer workshop," in Automatic Speech Recognition and Understanding Workshop. ICASSP, 2005.

[8] S. Yoon, M. Hasegawa-Johnson, and R. Sproat, "Landmark-based Automated Pronunciation Error Detection", Proceedings of Interspeech 2010 pp. 614-617.

[9] S. Abdou, S. Hamid, M. Rashwan, A. Samir, O. Abdel-Hamid, M. Shahin, W. Nazih, "Computer Aided Pronunciation Learning System Using Speech Recognition Techniques", INTERSPEECH 2006 - ICSLP, Pittsburgh, PA, USA.

[10] S. Hamid (2005) Computer Aided Pronunciation Learning System using Statistical Based Automatic Speech Recognition. PhD thesis, Cairo University, Cairo, Egypt.

[11] S. Hamed, M. Rashwan, "Automatic Generation of Hypotheses for Automatic Diagnosis of Pronunciation Errors" First International Conference on Arabic Language Resources and Tools, Cairo, Egypt, 22-23 April 2004.

Performance Evaluations For A Computer Aided Pronunciation Learning System

Sherif Abdou
Department of Information Technology
Faculty of Computers and Information, Cairo University
Giza, Egypt
sabdou@rdi-eg.com

Mohsen Rashwan
Department of Electronics and Communication
Faculty of Engineering, Cairo University
Giza, Egypt
mrashwan@rdi-eg.com

Abstract— This paper describes a speech-enabled Computer Aided Pronunciation Learning (CAPL) system HAFSS©. This system was developed for teaching Arabic pronunciations to non-native speakers. A challenging application of HAFSS© is teaching the correct recitation of the holy Qur'an. To evaluate the performance of HAFSS system a new evaluation technique is introduced. The proposed technique evaluates the system by measuring the degree of usefulness of its feedback to learners. Evaluating CAPL systems by this means forces system designers to try to emphasize the system response for confident decisions and make general feedbacks, or no comments for non-confident decisions to reduce deceiving effect of inherent speech recognition systems limited accuracy. Automation of the evaluation process is vital due to complexity of CAPL systems and the existence for many tunable thresholds and parameters.

Keywords- Speech processing; pronunciation learning, Arabic Language learning; performance evaluation

I. INTRODUCTION

Computer Aided Pronunciation Learning (CAPL) has received a considerable attention in recent years. Many research efforts have been done for improvement of such systems especially in the field of second language teaching [1] [2]. A challenging application for CAPL is the automatic training for correct recitation of the holy Qur'an for Arabic speakers. In contrast to the foreign language training task, where a wide variety of pronunciations can be accepted by native speakers as being correct, the holy Qur'an has to be recited the same way as in the classical Arabic dialect and the tolerance for allowed variation is very fine.

Most of the systems currently available for teaching the Holy Qur'an [3] are of multimedia type where the user learns to read the Holy Qur'an by listening to and repeating after the tutor. In such systems the user listens to several variants of the recitation of the given verse where only one of them is correct. If the user succeeded to select the correct one it is assumed that he comprehended the current lesson. So these systems test user comprehension of recitation rules not his performance in applying them.

The "HAFSS© system [4] is a tool for teaching Recitation rules and automatic assessment of the recitation of the Holy Qur'an. The system accepts the user recitation of some practice sentences from the Holy Qur'an and then assesses the quality of the user's recitation. The system produces a detailed report that includes a list of feedback messages. These messages help the user to locate his pronunciation errors, if they exist, and also give him guiding instructions on how to overcome them. So the HAFSS system is interactive one compared to the previous similar systems that rely on the user himself to evaluate and compare his recitation with recitations of professional tutors.

The HAFSS© system uses a state of the art speech recognizer to detect errors in user recitation. To increase accuracy of the speech recognizer, only probable pronunciation variants, that cover all common types of recitation errors, are examined by the speech decoder. A module for the automatic generation of pronunciation hypotheses is built as a component of the system [5]. The automatic generation of pronunciation hypotheses is reached by deploying matching rules to detect pronunciation patterns and generate corresponding probable recitation errors. Also a phoneme duration classification algorithm is implemented to detect recitation errors related to phoneme durations [6]. The decision reached by the recognizer is accompanied by a confidence score to reduce effect of misleading system feedback to unpredictable speech inputs [7]. To raise the accuracy of the HAFSS© system its models can be adapted to match the user characteristics [8]. A speaker adaptation algorithm suitable for the pronunciation assessment problem was developed. The implemented algorithm uses speaker classification, speaker normalization and Maximum Likelihood Linear Regression (MLLR) adaptation algorithms [9].

The HAFSS system needed a suitable evaluation technique to check the effects of modification in any part of the system to achieve fine tuning for system parameters and as a measure of the system quality. As parameter tuning is a continuous and critical procedure to overall system performance; automating the evaluation process is crucial for system development. In order to investigate some previously used evaluation techniques we used the traditional technique that is based on comparison of the human experts transcription with speech recognizer selected pronunciation variant. This resulted in measuring the system accuracy in correct phoneme or word percentages. Human experts sometimes disagree on one judgment on a phoneme pronunciation which can also happen for the same expert in different sessions. The major cause of disagreement is that there is no sharp boundary separating the pronunciation variants, and pronounced sound sometimes lies between two probable pronunciation variants. Also over concentration on a fatal pronunciation mistake can make an expert disregard an

adjacent minor mistake. Though we found this disagreement is less than 3% of the evaluation database, when the system approaches high accuracy decisions -in some specific problems-, this disagreement percentage constitutes a considerable amount of noise added to the system evaluation. Also confidence measures used in the system enables the system to generate general and/or ambiguous feedbacks to the student that can't be directly compared to human experts' hard-decision transcriptions. Some other techniques are based on computing the correlation between human and computer ratings [10]. In this technique the system feedback is just a rating of the quality of pronunciation of the utterance so it is not suitable for our target system. In this paper we present a new evaluation technique, which tries to overcome these problems by measuring degree of usefulness of the system response to the user.

In the following sections of this paper, section II includes an overview of HAFSS© system and its user interface. Section III includes the system structure and its main modules. Section IV includes the performance evaluation and assessment results for the HAFSS© system. Section V includes the final comments and the planned enhancements for the HAFSS© system.

II. System Overview

The user starts with the enrolment phase. In this phase the user is required to read a short phrase that is an easy and common phrase for Qur'an readers, so we expect almost no pronunciation errors for that phrase. This initial phrase is used to select the nearest model to that speaker from the system inventory of models. This model is used to judge the quality of the adaptation data. Then the user reads several phrases to be used for adapting the system models. Only the parts of the phrases that are judged by the system to be free of pronunciation errors are used to adapt the models. After models adaptation the user is allowed to enter the Exercises phase where he/she can select a lesson to practice a specific Recitation rule. Fig. 1 displays a capture for the HAFSS© system exercises screen. For each training example the user first listen to the tutor reading then records his own reading. The system judge the user input and produce a feedback message that includes the hypothesized errors. The errors are located and highlighted with different colors. Also the feedback message is played as a recorded audio. In case the user reading includes more than one error, in a side panel the user see a list of all his errors and can select any of them to see the system feedback message for that error. The speech recognizer of the HAFSS© system associates each decision it makes with a corresponding confidence score that is used to choose suitable feedback response to the learner. When the system suspects the presence of a pronunciation error with low confidence score the system has some alternate responses:

1- Omit the reporting of the error at all (which is good for novice users because reporting false alarms discourages them to continue learning correct pronunciation).

2- Ask the user to repeat the utterance because it was not pronounced clearly.

3- Report the existence of an unidentified error and ask the user to repeat the utterance (which is better for more advanced users than ignoring an existent error or reporting wrong type of pronunciation error).

4- Report most probable pronunciation error (which if wrong can be very annoying to many users).

Figure 1. Figure.1: HAFSS© system Exercises Screen

The HAFSS© system is integrated in an educational package that includes some animations for teaching the ideal articulations for the Arabic phonemes set. Fig. 2 shows a capture for one of the HAFSS© system flashes.

Figure 2. Figure.2: HAFSS Pronunciation Teaching Flash

III. System Architecture

Fig. 3 Shows the block diagram of the HAFSS© system. It deploys Hidden Markov Model (HMM) based speech recognizer that segments the utterance under test according to reference HMM acoustic models, the learner adaptation data, and the pronunciation variants for the test sentence. The speech recognizer associates each decision it makes with a corresponding confidence score that is used to choose suitable feedback response to the learner. The system main blocks are:

1. **Verification HMM models**: Is the acoustic HMM models for the system.
2. Automatic Speech Recognizer (ASR): The decoder that recognizes the user input speech.
3. **Speaker Adaptation:** Is used to adapt acoustic models to each user acoustic properties in order to boost system performance. It uses speaker classification, Maximum Likelihood Linear Regression (MLLR) speaker adaptation algorithms and supervised incremental technique [9].

4. **Pronunciation hypotheses generator**: It analyzes current prompt and generates all possible pronunciation variants that are fed to the speech recognizer in order to test them against the spoken utterance. [3].
5. **Confidence Score Analysis:** It receives n-best decoded word sequence from the decoder, then analyzes their scores to determine whether to report that result or not. [4].
6. **Phoneme duration analysis:** For phonemes that have variable duration according to its location in the Holy Qur'an, this layer determines whether these phonemes have correct lengths or not. To overcome inter-speaker and intera-speaker variability in recitation speed that may mislead the phone duration classification module. An algorithm for Recitation Rate Normalization (RRN) was developed [3].
7. **Feedback Generator:** Analyze results from the speech recognizer and user selectable options to produce useful feedback messages to the user.

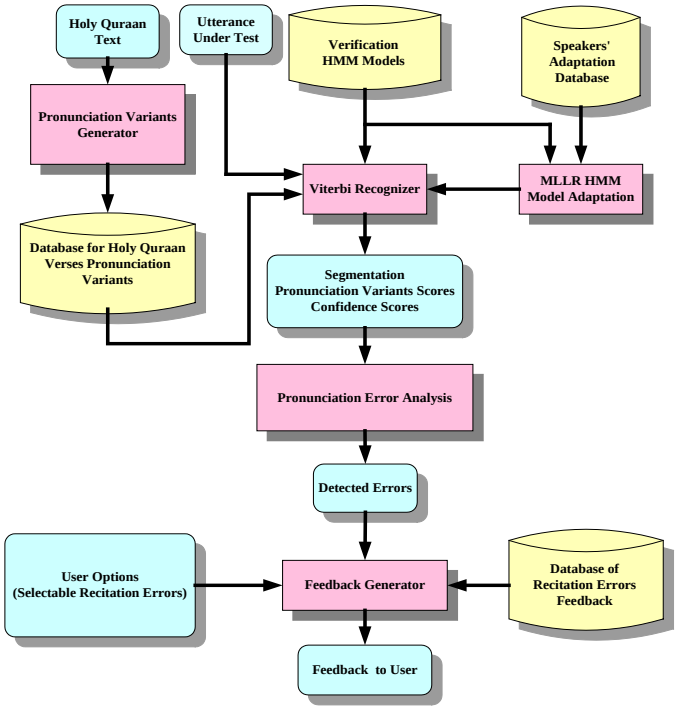


Figure 3. Block diagram of HAFSS©

IV. System Evaluations

Several assessments were performed to evaluate the correctness of the system feedback messages, the system performance compared to human references, the progress rate for the system users and the impact of using the system in learning environments.

A. *Evaluation of the system performance*

We asked 9 certified professional Qur'an reciters to read all the training examples included in HAFSS© with several trials for each example. They were asked to read the example correctly and also with pronunciation errors. The errors had to be among the possible errors for the Recitation rule under test for that example. The collected data from that test was 30317 utterances.

These utterances were evaluated by a number of language experts, and were labeled with the actual pronounced phonemes. Each expert was allowed to transcript the utterances in a separate session to avoid the possibility that his decision is affected by his colleagues' opinions. For ambiguous speech segments experts were allowed to write all acceptable judgments in their opinions. All experts' transcriptions were summed to produce a list of all the judgments accepted by the experts. Afterwards, a final group session was held where all experts discuss each error and they can agree on either to keep all the judgments or choose one or more of them, that's to correct any transcription errors that may be generated by them. This database was used to evaluate the system, by comparing the system responses with the experts' transcriptions. For that database, the experts' judgment had four possibilities:

1- Correct (accepted by all human experts).

2- Identified pronunciation error (all human experts reported the same type of error).

3- Not Perfect (human experts disagreed whether to reject or accept the pronunciation). That can happen when the pronunciation of a segment is not perfectly correct.

4- Wrong with unidentified error type (human experts agreed that a pronunciation error exists but disagreed on its type). That can happen when the user makes complex or undocumented errors.

The HAFSS© system judgment is one of four possibilities:

1- Correct pronunciation.

2- Pronunciation error with specified error type.

3- Unknown whether correct or wrong (repeat request based on confidence score).

4- Error with an unidentified error type.

Table (I) includes the results for that experiment.

TABLE I. HAFSS© SYSTEM PERFORMANCE

		Human judgement				
		Correct	Wrong	Not Perfect	Wrong with Unidentified Error Types	Total
System Judgement	Correct	80.89%	1.43%	1.07%	0.00%	83.39%
	Wrong with same error type	0.00%	4.29%	0.18%	0.00%	4.46%
	Wrong With Wrong Error Type	0.00%	0.18%	0.00%	0.18%	0.36%
	Repeat Request	8.75%	1.96%	0.71%	0.00%	11.43%
	Wrong with Unidefined Error	0.00%	0.36%	0.00%	0.00%	0.36%
	Total	89.64%	8.21%	1.96%	0.18%	100.00%

As we see in table (I), for correct speech segments the system yielded "Repeat Request" for about 9.7% of the total correct words. That is because they had low confidence below the computed threshold, and the system gave a repeat request to avoid the possibility of false alarms. For wrong speech

segments which constitute 8.2% of the data, the system correctly identified the error in 52.2% of pronunciation errors, reported unidentified errors for 4.4% and gave "Repeat Request" for 23.9% of the errors. The system made false acceptance of 17% of total errors. The HAFSS© system managed to give the correct feedback message in 84% of the test set.

B. Evaluation against human experts

In this assessment we considered HAFSS© system as an expert that competes with human experts for the task of teaching Recitation rules. We selected 300 utterances from our dataset representing the recitations of users of different gender, age and proficiency combinations. These utterances were evaluated by four human experts and HAFSS©. We measured the percentage of agreement, the utterances that received the same judgment, between each human expert and each one of the other three human experts. Also we measured the percentage of agreement between HAFSS© and each one of the four human experts. Results are shown in Table (II).

TABLE II. AVERAGE PERCENTAGES OF AGREEMENTS

Expert 1	Expert 2	Expert 3	Expert 4	HAFSS©
78.8%	77.2%	81.0%	78.7%	81.5%

Also we measured the percentage of disagreement between a human expert and the common opinion, utterances that got the same judgment, of the other three human experts. We measured the same percentage between HAFSS© and the common opinion of the four human experts. Results are shown in Table (III).

TABLE III. AVERAGE PERCENTAGES OF DISAGREEMENTS

Expert 1	Expert 2	Expert 3	Expert 4	HAFSS©
4.8%	7.8%	6.8%	2.0%	4.1%

From the results of tables (II) and (III) we can see that the performance of HAFSS© system is comparable to the performance of a human expert for the Recitation rules teaching task.

C. Users benefit from the system

As the system target is to be used as a self-learning tool, we designed this experiment to evaluate the user benefit from the system. We selected 10 beginner students in a class for teaching Recitation rules. We measured the performance of those students using a pre-test that included three sections for comprehension, listening and reading skills for applying Recitation rules. We used a post-test to measure there performance after using HAFSS© to practice Recitation rules for one hour and measured their performance again after another hour of practice. The results of that experiment are shown at Fig. 4.

We can see that the performance of the users has improved significantly by more than 40% relatively after using HAFSS© for just one hour. After using HAFSS© for two hours the users' performance has improved by more than 70% in the most problematic rule in the test. This result shows that the HAFSS© system is useful and effective in teaching recitation rules especially for beginner users who show a dramatic improvement in performance.

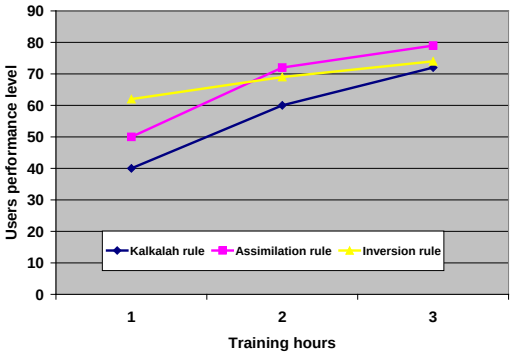


Figure 4. Figure (4) Users Performance Improvement

D. Using HAFS©S system in a recitation learning class

We evaluated the effect of the HAFSS© system when used in a traditional class for teaching Recitation rules. We selected two groups of students at the Schools of Riyadh at Saudi Arabia, each group consisted of 21 students. All the selected students were from the same class and can be considered educationally at the same level. We measured the performance of these two groups using a pre-test. The two groups were enrolled in a 4 weeks class to learn a subset of Recitation rules. The first group used only traditional learning of a human tutor who explained the Recitation rules to them and verified their recitations for the practice examples. The same tutor taught the Recitation rules to the second group but they used HAFSS© system to verify their recitation. The second group practiced with HAFSS© system without supervision from a Recitation tutor but they were allowed to practice for longer time than the first group. At the end of the class we measured the performance of the two groups using a post-test. The results of this evaluation are shown at Fig. 5.

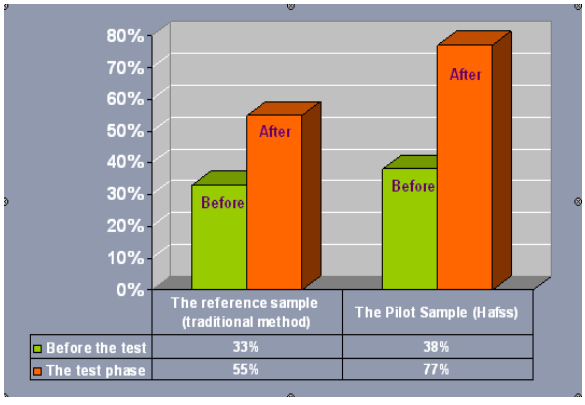


Figure 5. Figure .5: System Evaluations at Schools of Riyadh

The average pre-score for the traditional learning group was 33% and has improved after the Recitation class to 55%. The average score for the HAFSS© group was 38% and has improved after the class to 77%. This result shows that the percentage of improvement in students' performance using traditional learning is about 65%. While the students used HAFSS© system to practice recitation rules by themselves after

class, their performance has increased with 105% which is a significant amount of improvement.

V. CONCLUSIONS

In this paper we introduced the HAFSS© system. This system is a Computer Aided Pronunciation Learning (CAPL) system that is used to teach recitation of the Holy Qur'an. The system analyzes the user reading of a given verse from the Holy Qur'an then assesses the quality of the user recitation and produce feedback messages to help him locate his mistakes and overcome them. The HAFSS© system proved to be a useful one for the challenging task of automatic training for the correct recitation of the holy Qur'an for Arabic speakers. It not only helps students to learn how to recite the holy Qur'an but also helps them to correct their mistakes in formal Arabic pronunciation. The HAFSS© system is not designed to be a total substitute for a teacher but can be an effective assistant. We found that the HAFSS© system is more useful for beginner students than the advanced ones. Usually those beginner students are the ones who need the maximum care and effort from teachers. The HAFSS© system can help to free teachers' time by enabling students who need more individualized instruction to work independently with effective learning tools on a computer, while other students in the classroom receive more interaction and attention from the teacher. Our future work on HAFSS© will focus on using discriminative training techniques to improve the discrimination between some confusable pronunciation alternatives. Also we will try to improve the confidence score by using some articulation features. The system can be extended to be used in memorization of the Holy Qur'an.

ACKNOWLEDGMENTS

Special thanks are posed to The Research & Development International Company (RDI®) for its support of the pioneer application of CAPL technology in holy Qur`an recitation learning. We must also mention valuable efforts of speech technology and linguistic support teams at RDI®.

REFERENCES

[1] H. Franco et al., "The SRI EduSpeak system: Recognition and pronunciation scoring for language learning", Proc. of InSTIL, Scotland, 123-128, 2000.

[2] J. Mostow, et al., "Evaluation of an automated Reading Tutor that listens: Comparison to human tutoring and classroom instruction". Journal of Educational Computing Research, 29(1), 61-117, 2003.

[3] S. Hamid "Computer aided pronunciation learning system using statistical based automatic speech recognition". PhD thesis, Cairo University, Cairo, Egypt, 2005.

[4] S. Abdou, S. Hamid, M. Rashwan, A. Samir, O. Abdel-Hamid, M. Shahin, W. Nazih, "Computer aided pronunciation learning system using speech recognition techniques", INTERSPEECH 2006 - ICSLP, Pittsburgh, PA, USA.

[5] S. Hamid and M. Rashwan, "Automatic generation of hypotheses for automatic diagnosis of pronunciation errors" Proceedings of NEMLAR International Conference on Arabic Language Resources and Tools, pp. 135-139, Sep. 22-23, Cairo, Egypt, 2004

[6] A. Anastasakos, R. Schwartz, H. Shu, "Duration modeling in large vocabulary speech recognition" Proc. ICASSP, Vol. 1,628-631, 1995

[7] D. A. Williams, "Knowing what you don't know: roles for confidence measures in automatic speech recognition", Ph.D. thesis, Department of Computer Sciences, University of Sheffield, Sheffield, United Kingdom, 1999.

[8] T. Kosaka, and S. Sagayama, "Tree–structured speaker clustering for fast speaker adaptation", proceedings of the 1994 IEEE International Conference on Acoustics, Speech and Signal Processing 1, 245–248. IEEE, New York.

[9] A. Samir, S. M. Abdou, A. H. Khalil, M. Rashwan, "Enhancing usability of CAPL system for Qur'an recitation learning", INTERSPEECH 2007 - ICSLP, Antwerp, Belgium.

[10] L. Neumeyer, H. Franco, M. Weintraub, and P. Price, "Automatic text-independent pronunciation scoring of foreign language student speech. Proc. Proc. Int. Congress on Spoken Language Processing (ICSLP), 1996.

A Game on Pronunciation Feedback in Facebook

Nikos Tsourakis
ISSCO/TIM/FTI
University of Geneva
Geneva, Switzerland
Nikolaos.Tsourakis@unige.ch

***Abstract*—The proliferation of social networking services has revolutionized the way people around the world interact and communicate. The need to overcome language barriers that impose hurdles to this human communication is more acute than ever. Computer Assisted Language Learning systems try to bridge the gap between the need and the availability of language services in education. The convergence of both worlds is an obvious choice. In this work we extend a successful CALL platform and integrate it to Facebook, the most popular social network. After creating a pronunciation feedback game we investigate the interaction patterns of four users and present some preliminary results from ongoing work.**

Keywords-component; Pronunciation Feedback; Computer Assisted Language Learning; Social Networks

I. INTRODUCTION

Social networks are a useful tool to study social relationships and social phenomena through the connections among individuals instead of examining the properties for each one of them. It has been shown that all people are connected to one another by an average of "six degrees of separation" (your friend is one degree away from you) [1]. A recent study [2] reports that the average separation distance in Facebook, the most popular social network, is even less. After performing a world-scale social-network graph-distance computation they have shown that this number has been reduced to 3.74 "degrees of separation". The flow of information in our social entourage is constant as what we do and think has an impact as far as our friends' friends' friends, following the "three degrees of influence rule" [3].

Little attention has however been paid to exploring the full potential of integrating CALL systems in social networks. Benefits like performance feedback by your friends and the ability to comment, to request help, or to share content, would offer a more pleasant and engaging experience to learners. In this work we have integrated our CALL-SLT system into Facebook, the most popular social network service with more than 800 million active users. CALLSLT [4] is a spoken conversational partner designed for beginner- to intermediate-level language students who wish to improve their spoken fluency in a limited domain. It offers about a dozen combinations of L1s and L2s and is freely available for use (cf. callslt.org for detailed instructions).

CALL systems that rely on the output of the ASR to assess language skills engender the risk of accepting a sentence when in reality the pronunciation was incorrect (false positive) or rejecting a correctly pronounced sentence (false negative). This can increase the confusion of users concerning their pronunciation competence. Different tools for pronunciation training for second language learning try to alleviate this problem by either examining the production of speech [5] or by identifying confusable contexts [6].

This paper presents some early results of a work in progress after designing a pronunciation feedback game in Facebook. The idea behind the game was simple. Users spoke sentences in L2 (French) and acquired a pronunciation score in a scale between 0 (non-native) to 100 (native). Their score was juxtaposed with the one of their Facebook friends that had also used the system. We tried to investigate possible implications of this interaction scheme and to extract interaction patterns.

The rest of the paper is organized as follows. Section 2 describes the CALL-SLT system, and Section 3 the design of the experiment. Section 4 presents some initial results. The final section concludes.

II. THE CALL-SLT SYSTEM

CALL-SLT [4] is an Open Source speech-based CALL application for beginning to intermediate-level language students who wish to improve their spoken fluency. The system is deployed in the Web using a server/client architecture as described in [7]. Most processing, in particular speech recognition and language understanding, occurs on a remote server. The core idea is to give the student a prompt, formulated in their own (L1) language, indicating what they are supposed to say; the student then speaks in the learning (L2) language, and is scored on the quality of their response.

A. Presentation of Prompts

One way that CALL-SLT differs from other work (e. g. [8]) is in its presentation of prompts to the students. Instead of giving students complete prompts in their own language (the L1), our system uses interlingua representations in L1. In this way we avoid the undesirable effect of tying the language being studied (the L2) too closely to the L1 in the student's mind. In this work, the interlingua is realized in a telegraphic textual form but it is possible to produce graphical and video realizations of it without changing the underlying architecture. We support forms for different L1s, including English, French, Japanese, Chinese and Arabic.

The system is loaded with a set of possible prompts that represent the target content for a given lesson. Each turn starts

with the student asking for the next prompt. The system responds by showing a surface representation of the underlying interlingua for a target L2 sentence. For example, a student whose L1 is French and whose L2 is English might be given the following textual prompt:

COMMANDER DE_MANIERE_POLIE SALADE

Valid responses to this prompt would "I would like a salad", "Could I have the salad?", or simply "A salad, please"; the grammar supports most of the normal ways to formulate this type of request.

B. Pronunciation Module

In order to test our ideas in practice, we implemented a module that could elicit, in a simplified way, the pronunciation competence of each user. Each input sentence by the learner was assessed with respect to some reference utterances from native speakers. Specifically, we utilized data from 6 native French female subjects and from 6 female intermediate-level language students. Each one of them provided 30 sentences; half of the subjects in each group were used for training the module and the other half for testing it. After normalizing the waveforms, we used the software program praat [9] to extract the mean pitch, the mean intensity and the mean of the first-to-fifth formant frequencies of each utterance and used different combinations of them as our feature space.

Support Vector Machines have proven effective in a wide range of classification tasks, and were also chosen as the candidate method for our pronunciation module. We experimented with different combination of features and results of the SVM method (polynomial kernel and trade-off between training error and margin 5000) using the WEKA Toolkit [10] are shown in Table 1. The dyad of mean intensity and mean fifth formant provided the lowest error rate (98.88%), comparable with the result obtained using as feature the mean of the fifth formant (98.33%). When the mean intensity was chosen as the single feature, the correct classification reached to 83.33%, This suggests that non-native subjects did speak less loud, an indication of feeling less confident. The role of second (F2) and third (F3) formants frequencies for foreign accent determination have been reported in previous studies [11]. In our case however the mean value of formants was calculated for the whole spoken sentence and not for isolated phonemes, something that might explain why we did not encounter improved performance using the F2 or the F3.

The binary output of the classifier (native/non-native) was a restrictive factor for our experiment as ideally we would like a specific score for the input sentence between 0-100. In order to alleviate this deficiency, the module randomly produced a score between 30%-70% when the subject was classified as non-native and between 70%-95% when she was considered as native (Figure 1). The strategy of using narrow ranges of scores was imposed in order to avoid confusion when users uttered a sentence in a similar way and would expect a similar if not identical score. The module, bundled with praat and WEKA, constituted yet another step in the processing chain of CALL-SLT online system, with an average overhead latency of around 300 msec.

TABLE I. CLASSIFICATION ERROR (PERCENTAGE) OF USERS' NATIVENESS / NON- NATIVENESS

Features	Mean of: Pitch (P), Intensity (I), Formant (F1-F5)			
	Correctly Classified	Precision	Recall	F-Measure
P	61.11%	63.20%	53.30%	57.80%
I	83.33%	76.80%	95.60%	85.10%
P-I	83.88%	79.60%	91.10%	85.00%
P-I-F1	84.44%	79.80%	92.20%	85.60%
P-I-F2	83.33%	78.80%	91.10%	84.50%
P-I-F3	84.44%	79.20%	93.30%	85.70%
P-I-F4	96.11%	97.70%	94.40%	96.00%
P-I-F5	**98.88%**	**100.0%**	**97.80%**	**98.90%**
P-I-F1-F2	83.33%	78.80%	91.10%	84.50%
P-I-F1-F2-F3	84.44%	81.00%	90.00%	85.30%
I-F5	98.33%	100.0%	96.70%	98.30%
F5	**98.33%**	**100.0%**	**96.70%**	**98.30%**

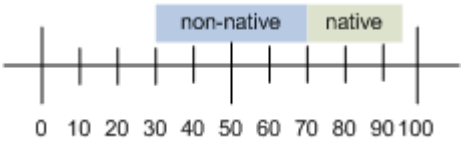


Figure 1. Score range for the native/non-native target groups

III. EXPERIMENTAL DESIGN

According to previous studies an opponent makes players more motivated and focused during a programming course [12] or a translation game [13], something that improves the effectiveness of the game per se. We therefore organized our experiment as a pronunciation game where social contacts constitute the potential opponents of users.

For each subject we constructed her contact's network as shown in Figure 2. The topology of the network was a concentric structure of two levels comprised of friends (one level) and friend's friends (two levels). There was also a third level with contacts outside user's network. We recruited subjects from our friends list in order to be able to utilize their social entourage. To avoid biases due to stereotypes all opponents were chosen from the same ethnic group (Greeks) and gender was approximately balanced. Moreover we picked image profiles with faces that didn't expose strong emotions. All participants had intermediate-level French language skills.

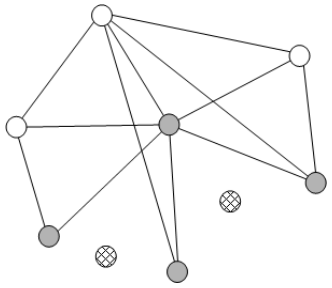


Figure 2. Topology of the contacts network. All nodes participated in the first round of the study (data collection) and only the grey modes in the second one (testing). Grid-filled nodes represent not connected subjects

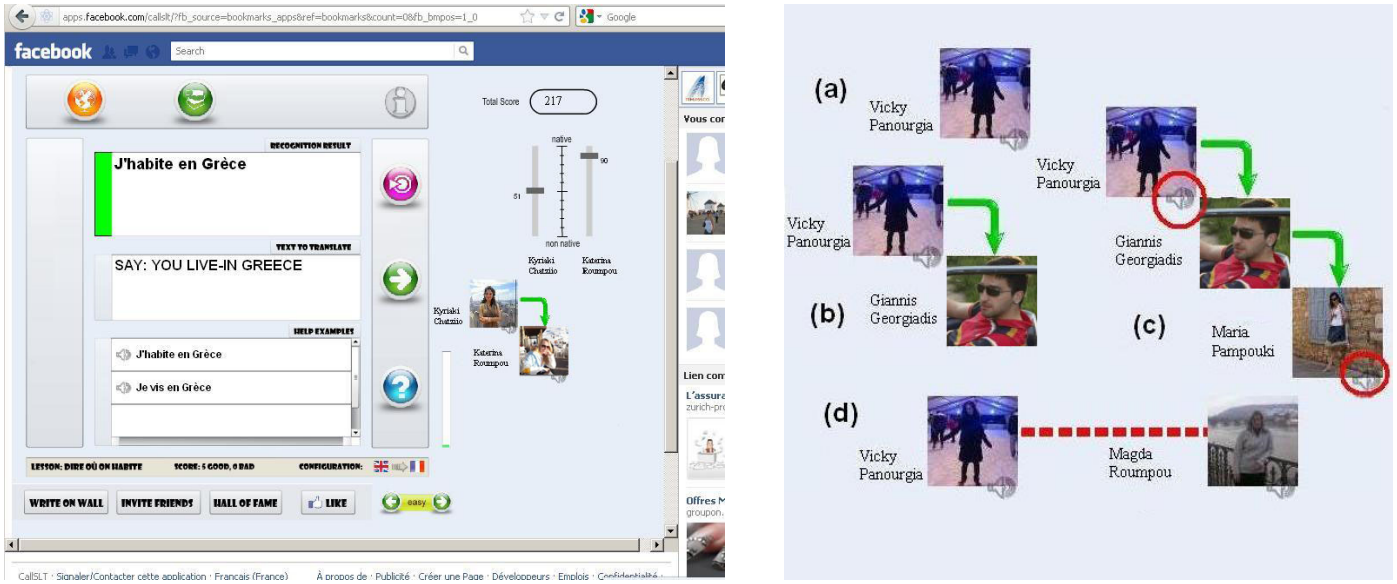


Figure 3. CALL-SLT as a Facebook application (left). On the left side the middle pane shows the prompt; the top pane, the recognition result; the bottom pane, text help examples. Pronunciation related widgets are presented on the right side. From bottom down, total score, pronunciation scale, contact connections. Presentation of the connections patterns (right). Alone (a), friend (b), friend of friend (c) not-connected to the subject (d). A speaker icon in the downright corner of the image signifies that the audio file is available for listening

The study was presented to the subjects as if we wanted to test a new feature, namely our new pronunciation module. The game was split into two rounds. During the first one all participants were asked to contribute to the database of sounds by using the application as they wished. Interacting with the standard system (no pronunciation assessment) was an essential exercise to familiarize themselves with the offered functionalities. Two weeks later the system was ready to be used for assessing users' pronunciation skills.

The interaction with the application is performed as follows. The prompt is presented in the middle textbox of Figure 3 (left). The user decides what she is going to say, presses the "recognize" (purple) button, and speaks. She may ask for acoustic help at any time by clicking the "help" (blue) button. After each turn the pronunciation score for the user and the one of the opponent are presented side by side in two vertical slide bars. We also animate the movement of the two sliders from the lower level (non-native) towards the upper one (native) to grab user's attention and focus on competition. The connection between the user and the opponent is presented as a cascaded sequence of profile images. The green arrow signifies connection, whereas the red dotted line the opposite (Figure 3(right)). For presenting scores and connections we considered the following:

- Results were offered only if the speech recognition was successful.
- Users were randomly exposed to four patterns in the same frequency, namely "Alone" (no comparison with a contact), "First" (comparison with a first level contact), "Second" (comparison with a second level contact) and "No Contact" (comparison with a not-connected subject).
- Users could repeat the same sentence multiple times while the opponent and his score remained the same.
- After a repetition of the same sentence the new user's score was in the range ± 15 of the score obtained in the previous turn, given that the classifier provided the same class (native or non-native).
- The total score was calculated as a sum of the pronunciation score at each turn and the game ended when it reached to 2500 points.
- The opponent's score using recognized waveforms was a random number between 15%-95%.

Our intention was not to impose direct competition among users by requesting the repeat of the same phrase if the subject got lower score compared to the one of the opponent but to let them decide when to do so. They had also the opportunity to listen to the opponent's waveform by clicking on his profile image or listen to their own last recorded utterance by clicking on their photo.

IV. Preliminary Results & Discussion

In this section we will provide some preliminary results obtained from 4 female users that interacted with the system. This first analysis focused on questions like: "Do users decide to repeat a prompt after they see the score of their contact?", "Does the difference in score matter?", "Is the distance of the contact important?", "Do users listen to the prompts of their opponents? ", etc. The results presented in Table 2 provide a first insight to the aforementioned questions.

As already mentioned the experiment was designed in a way so that participants would be equally exposed to the same connection patterns. In the "*Exposed to …*" rows we can observe that this was more or less accomplished. Each subject seems to have followed her own interaction style, which provides an initial distinction of user types:

TABLE II. RESULTS OF THE INTERACTION FROM 4 USERS

	User 1	User 2	User 3	User 4
Exposed to "Alone"	8	6	8	11
Exposed to "First"	6	6	9	11
Exposed to "Second"	6	8	9	12
Exposed to "No Contact"	8	6	9	12
Repeats after "Alone"	37.5%	66%	37.5%	9.09%
Repeats after "First"	50%	33.33%	11.12%	18.18%
Repeats after "Second"	37%	50%	44.45%	0%
Repeats after "No Contact"	25%	33.33%	22.23%	0%
Repeats after lower score	79.31%	65.38%	33.33%	5%
Repeats after higher score	0%	50%	28.57%	6.67%
Listen "Alone" prompt	0%	0%	12.5%	90.9%
Listen "First" prompt	0%	50%	66.67%	72.72%
Listen "Second" prompt	0%	0%	55.56%	58.34%
Listen "No Contact" prompt	0%	16.67%	22.23%	58.34%
Average prompt repetition	0	3	7.11	1.63

Competitive (User 1). The real incentive for her is the score balance. 79% of the times she got a lower score than the opponent's, she decided to retry. She feels so confident that she can do better by retrying that she doesn't need to listen to waveforms, even her own.

Experimenter (User 2). This subject uses all the functionalities offered by the system without explicitly focusing on one aspect of it.

Scholastic (User 3). On average this user listened to each contact's prompt 7.1 times, probably due to curiosity and in order to compare. She was not motivated too much to repeat the same sentence.

Social Spectator (User 4). The user doesn't show any signs of competition, which is why she rarely repeats a prompt. On the other hand, she exposes strong interest in listening to the prompts of her contacts (1.63 times).

In the previous categorization there is one to one association between user types and the four users. This distinction is our recommendation; it is indicative and by no means exhaustive. It might be useful to others intending to implement a similar protocol. More data is required to determine behavioral patterns and to form better distinctions.

A comment expressed by all participants was about the sensitivity of the pronunciation module. Their concerns were not related to its randomness, a fact they were not aware of, but mostly that they might not receive the same score when the sentence was spoken in the same way. Moreover there were times when it was not obvious what made an opponent's score better. Subjective self-assessment in language proficiency is always an issue even if a real pronunciation analysis takes place. Providing a cumulative score on pronunciation competence without specifying problematic regions in the spoken utterance would inevitably raise objections. In our case a more appropriate solution might be the substitution of the fine grained scale of the sliders with a scale of only 3-4 levels. The development of a more sophisticated pronunciation module would permit exploring the full potentials of our ideas.

V. CONCLUSIONS

In this work we tried to design a pronunciation game on Facebook and investigate the effects of integrating a CALL system with a social network. A core contribution is the addition of an element of motivation by involving another player/learner.

There were some inherent deficiencies in the game design however. Experimental subjects were recruited from the author's contact list in order for us to exploit their social entourage. An additional requirement was that they had an intermediate level in the French language. Both restrictions limited the study to few participants.

Finally, when deploying applications in social networks issues related to privacy must be addressed. During the recruitment phase a fifth subject politely rejected to contribute due to the fact that our Facebook application specifically requested access to her friends list.

REFERENCES

[1] P. S. Dodds, R. Muhamad and D. J. Watts, "An Experimental Study of Search in Global Social Networks," Science 301 (2003): 827-29.

[2] L. Backstrom, P. Boldi, M. Rosa, J. Ugander, S. Vigna, "Four Degrees of Separation," WebSci 2012.

[3] N. A. Christakis and J. H. Fowler, "Connected: The Suprising Power of Our Social Networks and How They Shape Our Lives," Little Brown: New York, 2009.

[4] P. Bouillon, S. Halimi, M. Rayner, N. Tsourakis, "Evaluating a web-based spoken translation game for learning domain language," Proceedings of INTED, Valencia, Spain, 2011.

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

[6] O. Saz, M. Eskenazi, "Identifying Confusable Contexts for Automatic Generation of Activities in Second Language Pronunciation Training," Proceedings of the SLaTE Workshop, Venice, Italy, 2011.

[7] M. Fuchs, N. Tsourakis, M. Rayner, "A Lighweight Scalable Architecture For Web Deployment of Multilingual Spoken Dialogue Systems," Proceedings of LREC 2012, Istanbul, Turkey.

[8] C. Wang and S. Seneff, "Automatic assessment of student translations for foreign language tutoring," Proceedings of NAACL/HLT 2007.

[9] P. Boersma and D. Weenink. Praat: Doing phonetics by computer. http://www.praat.org/.

[10] M. Hall, E. Frank, E. G. Holmes, B. Pfahringer, P. Reutemann, I. Witten, "The WEKA Data Mining Software: An Update," SIGKDD Explorations, Volume 11, Issue 1, 2009.

[11] L. M. Arslan, J. H. L Hansen, "A Study of Temporal Features and Frequency Characteristics in American English Foreign Accent," The Journal of the Acoustical Society of America, July 1997.

[12] R. Lawrence, "Teaching data structures using competitive games," Education, IEEE Transactions on, vol. 47, no. 4, pp. 459 – 466, 2004.

[13] W. Ling, I. Trancoso, R. Prada, "An Agent Based Competitive Translation Game for Second Language Learning," SLaTE Workshop, Venice, Italy, 2011.

Underspecification in Pronunciation Variation

Mark Kane, Zeeshan Ahmed and Julie Carson-Berndsen

CNGL, School of Computer Science and Informatics, University College Dublin, Ireland

mark.kane@ucdconnect.ie
zeeshan.ahmed@ucdconnect.ie
julie.berndsen@ucd.ie

***Abstract*—This paper presents a technique whereby underspecification of phonetic units offers additional robustness yielding an increase in performance in a pronunciation variation task; spoken term detection. The approach is based on first recognising phones from a spoken query and then dynamically matching these spoken query phones to an unknown utterance of recognised phones. The experiment describes how underspecifying a phonetic unit to create an archiphoneme that belongs to a broad phonological group aides this task by tackling pronunciation variation caused by a speaker and inaccuracies in the phonological unit classification, such as substituted phones whilst deletions and insertions are implicitly tackled using a Forward-Backward search. Additionally, a hybrid search using the Forward-Backward search as an adjustment window is used to explicitly tackle deletions and insertions using underspecification. The results were found to be comparable to the Forward-Backward search.**

I. Introduction

In many forms of speech processing, accounting for *pronunciation variation* is an important research topic. This paper focuses on using underspecification to tackle pronunciation variation in the task of Spoken Term Detection. The specific application of spoken term detection for this work is intended as a front-end to a pronunciation learning system that would allow a student to speak a term and the system to return a list of utterances where that term exists in a database allowing the student to listen to a term in different contexts. This research is part of our web based spoken language coach called MySpeech, a Centre for Next Generation Localisation (CNGL) demonstrator system [1], [2].

One underlying problem of spoken term detection is the ability of a system to handle pronunciation variation. Variation can be attributed to several factors such as accent, gender, geographical location and emotional state to name but a few and that is before ones non-nativeness is taken into account. Furthermore variation can also be incurred from the automated process of detecting the spoken term, such as phone substitutions, insertions and deletions for phone matching based systems. Bi-gram and tri-gram phone confusions are accounted for in [3], where dynamic programming is used for sequence matching, yielding substantial gains.

In [4], a dynamic match phone-lattice search is used to compensate for phone substitutions, insertions and deletions in a computationally cost effective manner. Phone confusion probabilities are also used in [5] accompanied by other knowledge resources such as *manner* and *voicing* features which are used for pruning and rescoring in a keyword filler based approach.

Query-by-example techniques in spoken term detection are advantageous in; low-resource, notable phonetical variance between recognition system and accent of user under test and OOV situations as described in [6] and [7].

The approach presented in this paper applies the ethos of [8], to integrate additional knowledge sources in an effort to create a knowledge-rich data-driven system in comparison to a purely statistical learning technique. Hidden Markov models are first used to decode a spoken term query and then a speech utterance into phones. An approach that goes beyond the first best alignment path is described in [9] and uses a multi-lattice alignment process yielding encouraging results for spoken keyword spotting. In this paper the alignment of the phones of the spoken term with the phones which are recognised in the utterance is implemented using dynamic programming where the minimum of either the Forward or Backward search is used where an insertion or deletion may affect one path more so than another.

In an effort to account for phonological based lexical variation [10] referred to in [11] as *individual error detection*, the population of the difference matrix is based on assigning difference scores to phones belonging to the same archiphoneme often referred to as a Broad Phonetic Group (BPG), however here it is aptly named *broad phonological group* (BPG).

Broad phonetic groups described in [12] state that 75% of all misclassified frames were found to belong to the same group, where phones within these groups would have similar characteristics. These groups permit allowable phone substitutions in phone based spoken term detection. The phone assignment to BPGs of this paper is based on the idea that particular underlying articulatory features, features which appear simultaneously within the phone, can either be *on*, *off*, *unspecified* or *unused* as recently detailed in [13].

In an effort to explicitly tackle deletions and insertions beyond the Forward-Backward approach, a hybrid search is evaluated using underspecification. This approach modifies the forward and backward path in the form of an *adjustment-window*, where the best path is found within this windowed search space and thus any alignment errors, found at the end of a Forward or Backward search caused by insertions or deletions, are ignored.

The experiment presented in this paper highlights the improvements made by the dynamic approach by accounting for such pronunciation variations using underspecification against a baseline that does not use underspecification.

The remainder of the paper is structured as follows. Section II introduces underspecification and section III details the experiment that is carried out. The results of the experiment are discussed in section IV and directions for future work are highlighted in section V. Finally, conclusions are drawn in section VI.

II. Underspecification

Underspecification results in the classification of a phonological unit into a BPG whereby a specific phonetic feature is considered marked or unmarked, depending upon the markedness criteria. The underspecification methodology used to distinguish these groups is based on defining a phone with respect to its Articulatory Features (AFs) where a phone is considered fully specified if in the same interval it has a value for all AFs associated with it. This paradigm builds on theories of autosegmental and articulatory phonology and is underpinned by phonological feature theory from *Trubetzkoy* and [14] to [15] and [16]. In the past two decades, they have been further developed for speech technology in [17], [18], [19], [20] and [21].

Recently, underspecification of articulatory based phones described in [22] shows an increase in performance in resolving conflicts in multiple source phone recognition. Other recent work describes how underspecification is used to define *BPGs* for the purpose of introducing *difficulty levels*, similar to that of a digital game, in a pronunciation learning environment [1] and clearly indicates that *BPGs* are effective in tackling pronunciation variation even for non-native speakers. The AFs used in this paper are primarily based on the International Phonetic Alphabet (IPA) [23]. Phone-to-AF mapping is also extensively described in [24] and [25]. In the latter, acoustic based features called *dynamic/static* are used to define the rate-of-spectral-change of a segment and are described here in the context of the classification of BPGs.

A. Broad Phonological Groups

BPGs offer the dynamic programming approach a way to compare two phone labels when these labels are different. By taking into account the canonical phonetic make-up of a phone, phones can be compared by underspecifying this phonetic make-up. In this approach articulatory features are used, where a phone is considered fully specified if it has all the canonical AFs associated with it. When comparing two phones, a phonetic difference score is assigned to phones belonging to the same BPG instead of a maximum difference score given if the two phone labels are different. The application of these BPGs is outlined in section III-A.

Additionally, vowel insertion associated with the substitution of syllabic pairs is also accounted for in these BPGs. An example is where the phone *el* is substituted with phone *l*. Both phones belong to the same broad phonological group so only a minor penalty is incurred. However looking at the broader context of this substitution, *l* may have been preceded by a vowel such as *ax* so therefore it is *ax l* that is substituted for *el*. In these circumstances no penalty is incurred for a vowel insertion. The complete set of BPGs implemented in this paper are described in Table I.

TABLE I
Broad phonological groups (BPGs)

BPGs	phonemes	description
#1	n nx	alveolar nasal/flap static
#2	n m	alveolar/bilabial nasal
#3	er axr	rhotic/retroflex
#4	th dh t d dx	dental/aveolar dynamic fricatives/plosives/flaps
#5	p b	minimal pair - voicing
#6	k g	minimal pair - voicing
#7	ch jh	minimal pair - voicing
#8	f v	minimal pair - voicing
#9	s z	minimal pair - voicing
#10	sh zh	minimal pair - voicing
#11	hh hv	minimal pair - voicing
#12	iy ix ih uw ux uh	vowel height - high
#13	ey ow eh ah ax	vowel height - mid
#14	ae ao aa	vowel height - low
#15	iy ih ey eh ae	front vowels
#16	ix ux ax axh ah ih	centre vowels
#17	uw uh ow ah ao aa	back vowels
#18	#1 en	syllabic pair
#19	m em	syllabic pair
#20	ng eng	syllabic pair
#21	r #3	syllabic pair
#22	l el	syllabic pair
#23	el [(#12 |#13 |#14) l]	syllabic pair - vowel insertion
#24	#2 [(#12 |#13 |#14) r]	syllabic pair - vowel insertion
#25	en [(#12 |#13 |#14) n]	syllabic pair - vowel insertion
#26	em [(#12 |#13 |#14) m]	syllabic pair - vowel insertion
#27	eng [(#12 |#13 |#14) ng]	syllabic pair - vowel insertion
#28	aw ay oy #12 #13 #14	diphthongs

III. Experiment

The experiment presented compares a spoken term detection without BPGs (baseline) to a spoken term detection with BPGs determined by underspecification where the phones of each query term for both approaches are hypothesised from a phone recogniser.

Whilst this work is intended as a front-end system to the pronunciation learning system described in [1] and [2], the evaluation of this approach does not include non-native speakers at this stage. The results and conclusions drawn from this paper are intended to be a baseline against non-native speakers in future work. The reasoning for this two stage approach is due to the fact that a native speaker using a CALL system, where the target language is the same as the users, should produce competent results or act as a threshold for non-native speakers.

In this experiment the phones that constitute the recognised spoken term are aligned with the phones recognised from an utterance. From this alignment process, a distance measure is calculated representing the similarity between the spoken term phones and the recognised utterance phones where the final output of the process is a ranked order of filenames/utterances and timing information based on their similarity where a student will then be able to listen and practice these utterances as illustrated in Figure 1.

From the corpus test set outlined in section III-B, the terms used for evaluation are created with respect to the following criteria; 1) all terms must appear at least twice and 2) each term

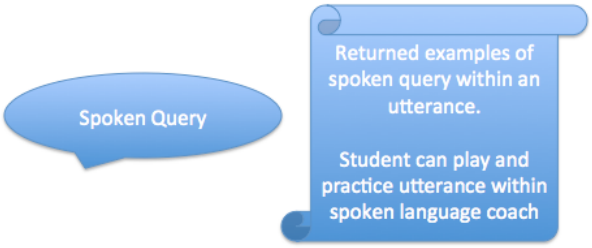


Fig. 1. *Application of spoken term detection*

is canonically made up of greater than five phones. In total this yields 273 words for evaluation where these terms occur 1649 times in a test set containing 1344 utterances (~1.5 hours). Each term occurs on average in approximately 6 utterances. A query term is taken from the test set where one instance of a term is chosen as described in [9] and the utterance to which it belongs is not used for evaluation.

This result is then evaluated using the average recall for all spoken terms at 5, 6*, 10, 15 and 20 false alarms where * is the average occurrence of a term in the test set.

A starting condition of this experiment is based on parsing the phones of an utterance recognised from a triphone recogniser described in section III-B. A *parsing window* moves across these phones where the spoken term detection process only tries to match the phones from the parsing window if the number of phones from this window pw lies within the range

$$(0.25 * N) < pw < (1.5 * N)$$

where N is the number of phones in a recognised spoken term. This starting condition avoids needless computation.

Once a parsing window of recognised phones is obtained, dynamic programming is used to align the recognised phones of the spoken term to the phones of the parsing window where the BPGs are used to populate the difference matrix in this process. This dynamic programming approach will now be explained in more detail. For the remainder of this section the recognised spoken term phones are called the *reference* and the recognised utterance phones from the parsing window are called the *test*.

A. Dynamic Programming

The first stage of this process is to populate a difference matrix $(N * M)$, where N is the number of phones in the *reference* and M is the number of phones in the *test*. The difference between each of the phones is determined using underspecification as noted in section II and it is these differences that populate the difference matrix. A minimum phonological-difference path is then found through the difference matrix using a *Forward-Backward and Hybrid search*.

- *Population of difference matrix*. Each phone within the reference and test are compared against each other whereby a resultant matrix of difference scores for each co-ordinate are found. If a phone is equal to that of another, the difference score is 0. If a phone is not equal but both belong to the same BPG, then the difference is 1. All other differences are set to 7 indicating a maximum difference.
  An exception to these population rules during search is found when either $j = N - 1$, in this case only one movement is allowed:
  $$D(N-1, i+1)$$
  or $i = M - 1$ where the only movement allowed is:
  $$D(j+1, M-1)$$
  where j represents the reference phone and i represents the test phone. When this exception within the difference matrix is found, a penalty of 7 is added to the phonological-difference path.
- *Forward-Backward path.* This approach finds the minimum of either the Forward or Backward search which is then used as the final phonological-difference score. The Forward path starts at $D(0,0)$ and finishes at $D(N-1, M-1)$. The Backward path starts at $D(N-1, M-1)$ and finishes at $D(0,0)$. This approach specifically handles deletions and insertions which can individually mislead both *Forward* and *Backward* searches.
- *Hybrid path*. This is achieved by using the *Forward* path and the *Backward* path as an *adjustment window* and then computing if and where it is effective to move from one path to another where a penalty of 7 is incurred for every row/column skipped between the two paths and are illustrated in Figure 2 (moving from one path down to another on the same column) and Figure 3 (moving from one path across to another on the same row). The concept of using an adjustment window, albeit calculated by other means, is described in [26].

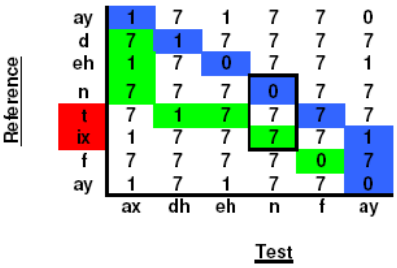


Fig. 2. *Example of Hybrid path overcoming deletions for the word 'identify'.*

B. Recognition System and Corpus

The TIMIT speech corpus [27] is used for training and testing in this experiment in order to compare results with [9]. This corpus consists of read speech spoken by 630 speakers of American English. The data is split into two sets; training and complete test set. The training set consists of 3696 utterances while the test consists of 1344 utterances. The SA data is not used in this paper. All plosive phones represented by a separate

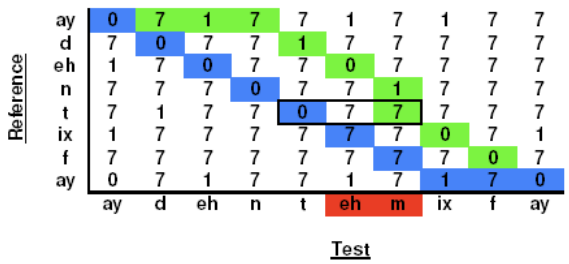


Fig. 3. *Example of Hybrid path overcoming insertions for the word 'identify'.*

closure and associated burst are merged into a single phone e.g. *dcl d* is relabelled as *d* where *dcl* on its own is relabelled as *d*. The HMM based speech recognition system used in this experiment is implemented with HTK [28]. The chosen form of parameterisation of a phone within an utterance is mel frequency cepstral coefficients (MFCCs), with their associated log energy and first and second order regression coefficients. Therefore every frame is represented by 39 coefficients. The MFCCs representing the phones are then used in the calculation of HMM models. The HMMs are context-dependent triphone models that were initially calculated by cloning and re-estimating context-independent monophone models. The triphones states were tied using phonetic feature decision trees for clustering. Each model is comprised of 5 states where only the centre 3 states are emitting. The decoding process is implemented with a tri-gram phone model. Finally, the number of components in each mixture is set to 8 as this was found to be the optimal number for the corpus.

IV. Results and Discussion

In this section the results for spoken term detection using different methods are presented and discussed. Primarily there are two different categories of evaluation, experimental results *without* (baseline) and *with* BPGs.

Two different search types are both implemented without and with BPGs; Forward-Backward and Hybrid search as discussed in section III. While both search types use dynamic programming and implicitly handle substitutions, insertions and deletions, the Hybrid search explicitly tackles insertions and deletions that can affect the path in which a dynamic process will move through a difference matrix.

A. Phone recognition

For a phone based spoken term detection, the phone accuracy of the recognition system has a cascade effect on spoken term detection results. This hierarchical approach causes hard decisions to be made at a level before the spoken term detection process. However, underspecification of the phone in the spoken term detection process goes beyond the one-to-one comparison of phones by designating phones to BPGs thus allowing a score to be calculated indicating the phonetic similarity of phones i.e. their phonological difference. For comparison, the initial phone (triphone) recognition accuracy of the system is 64.63% and similar to [9].

B. Without (baseline) and with BPGs

The spoken term detection results without and with BPGs are outlined in Table II. This approach uses the Forward-Backward search type and finds the minimal path between either the forward or the backward path through a difference matrix. In this table, recall results are given with respect to a different number of false alarms. Additionally, the recall for the number of false alarms equivalent to the average occurrence of terms in the test (six) is also shown. As can be seen from this table for both without (baseline) and with BPGs, as the false alarms increase the recall of the system increases as expected. At each false alarm, the recall is greater with BPGs than without where the *difference* between systems without and with BPGs is also shown. At six false alarms the increase in spoken term detection performance using BPGs defined using underspecification is 10.5%. Notably, the effect of using BPGs diminishes by 2.7% as the number of false alarms increases from 5 to 20.

Another method that is used for *spoken keyword spotting* is [9] which uses a multi-lattice alignment approach. The experiment presented there is also carried out on the same corpus and also examines words of phone length greater than 5 with a frequency of occurrence of at least two. However, their keyword test set is approximately only a quarter of the size of the number of keywords presented in this paper. Taking these considerations into account, the performance of this paper at 5 false alarms without BPGs is 55.9% and with BPGs is 66.9% in comparison to the performance of [9] at 5 false alarms is 50% even though the PER of this paper's experiment is lower by 1.97% in comparison to [9].

TABLE II
Forward-Backward search: Average recall at 5, 6, 10, 15 and 20 false alarms where * is the average occurrence of a term in the test set.*

Recall (%) @ false alarm			
false alarms	Recall with no BPGs	Recall with BPGs	difference
5	55.9	66.9	+11.0
6*	57.5	68.0	+10.5
10	62.1	71.6	+9.5
15	65.7	75.0	+9.3
20	68.9	77.2	+8.3

C. Hybrid search

In addition to the Forward-Backward search, an experiment based on a Hybrid search without and with BPGs is evaluated to see if an optimum path can be found within a difference matrix using the forward path and backward path as an adjustment window. The results as shown in Table III indicate that while the *difference* between the Hybrid search without and with BPGs is marginally greater than the Forward-Backward search, the overall performance of the Forward-Backward and Hybrid search with BPGs is practically the same. This indicates that insertions and deletions that affect the path within a dynamic process in a difference matrix are effectively caught within the Forward-Backward search and that the extra computational cost of

the Hybrid search is not warranted at this time (albeit open for further research). Future work may benefit from knowing the computational cost and their gains versus more heavily weighted computational methods such as Breath or Depth First search. To visually compare the Forward-Backward and Hybrid search without and with BPGs a ROC diagram is used, Figure 4. The clear indicator for increased performance is noted in the curves representing approaches using BPGs.

TABLE III
*Hybrid search: Average recall at 5, 10, 15 and 20 false alarms where * is the average occurrence of a term in the test set.*

Recall (%) @ false alarm			
false alarms	Recall with no BPGs	Recall with BPGs	difference
5	55.4	67.2	+11.8
6*	57.1	68.5	+11.4
10	62.0	72.1	+10.1
15	64.9	75.5	+10.6
20	68.6	77.4	+8.8

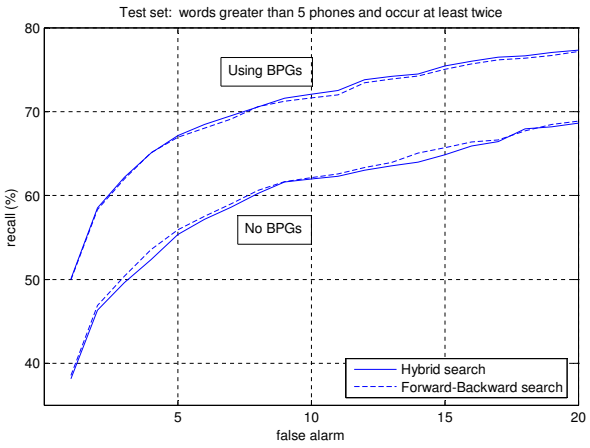


Fig. 4. *Recall versus false alarm: ROC curves for spoken term detection evaluation using different search methods with and without BPGs.*

V. Future Work

The next stage of this research is to evaluate the system using non-native students, where the standard of nativeness is set from this paper. Other future work will primarily be focused on using underspecification to extend the set of BPGs which attains optimum performance. Furthermore, difference matrix population will incorporate an increase in resolution (more fine-grained) beyond that of 0,1 and 7. Additionally, using an autosegmental approach with regards to articulatory features and syllable position will also be pursued.

VI. Conclusions

The experiment carried out in this paper demonstrates that the task of spoken term detection is vastly improved when pronunciation variation is accounted for in a dynamic programming environment. Specifically knowledge rich information of phonological substitutions is included where this knowledge rich information is based on grouping phones into broad phonological groups (BPGs) defined by underspecification that takes into account specific markedness of phonetic features such as the articulatory feature information of a phone. The results show that using BPGs increases the task's performance by 10.5% over a baseline that does not use BPGs. The experiment also involved identifying other insertions and deletions by finding a Hybrid path that consists partially of a Forward and Backward path where these two paths are used as an adjustment window for the search. This yielded a minor increase in performance in conjunction with BPGs.

VII. Acknowledgements

This research is supported by the Science Foundation Ireland (Grant 07/CE/I1142) as part of the Centre for Next Generation Localisation (www.cngl.ie) at University College Dublin. The opinions, findings and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of Science Foundation Ireland.

References

[1] Kane, M., Cabral, J., Zahra, A. and Carson-Berndsen, J., (2011). "Introducing Difficulty-Levels in Pronunciation Learning." Proceedings of the International Speech Communication Association Special Interest Group on Speech and Language Technology in Education (SLaTE), Venice.

[2] Cabral, J. P., Kane, M., Ahmed, Z., Abou-Zleikha, M., Székely, É., Zahra, A., Ogbureke, K. U., Cahill, P., Jiang, J. and Carson-Berndsen, J., 2012. "Rapidly Testing the Interaction Model of a Pronunciation Training System via Wizard-of-Oz." Proceedings of the eighth international conference on Language Resources and Evaluation (LREC).

[3] Chaudhari, U. V. and Picheny, M., 2007. "Improvements in phone based audio search via constrained match with high order confusion estimates." IEEE Workshop on Automatic Speech Recognition and Understanding. ASRU. Pages 665 - 670.

[4] Thambiratnam, K. and Sridharan, S., 2005. "Dynamic Match Phone-Lattice Searches For Very Fast And Accurate Unrestricted Vocabulary Keyword Spotting." IEEE International Conference on Acoustics, Speech, and Signal Processing,(ICASSP), Pages 465 - 468.

[5] Ma, C. and Lee, C. H., 2007. "A study on word detector design and knowledge-based pruning and rescoring." In InterSpeech, Pages 1473-1476.

[6] Hazen, T. J., Shen, W. and White, C., 2009. "Query-By-Example Spoken Term Detection Using Phonetic Posteriorgram Templates." IEEE Workshop on Automatic Speech Recognition Understanding (ASRU). Pages 421 - 426.

[7] Parada, C., Sethy, A. and Ramabhadran B., 2009. "Query-By-Example Spoken Term Detection For OOV Terms." IEEE Workshop on Automatic Speech Recognition Understanding (ASRU). Pages 404 - 409.

[8] Lee, C. H, 2004. "From Knowledge-Ignorant to Knowledge-Rich Modeling: A New Speech Research Paradigm for Next Generation Automatic Speech Recognition". InterSpeech, Plenary Paper.

[9] Lin, H., Stupakov, A. and Bilmes, J., 2009. "Improving multi-lattice alignment based spoken keyword spotting." IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP).

[10] Strik, H. and Cucchiarini, C., 1999. "Modeling pronunciation variation for ASR: A survey of the literature." Speech Communication, Volume 29, Pages 225 - 246.

[11] Maxine Eskenazi, 2009. “An overview of spoken language technology for education.” Speech Communication, Volume 51, Pages 832 - 844.
[12] Scanlon, P., Ellis, D. and Reilly, R., 2007. “Using broad phonetic group experts for improved speech recognition”. IEEE Transactions on Audio, Speech and Language Processing, vol. 15, no. 3, Pages 803 - 812.
[13] Bates, R. A., Ostendorf, M. and Wright, R. A., 2007. “Symbolic phonetic features for modeling of pronunciation variation.” Speech Communication. Volume 49, Issue 2, Pages 83-97 .
[14] Jakobson R., Fant, G. M. C. and Halle, M., 1952. “Preliminaries to Speech Analysis: The Distinctive Features and their Correlates.” MIT Press, Cambridge, MA, U.S.A.
[15] Chomsky, N. and Halle, M., 1968. “The Sound Pattern of English.” MIT Press, Cambridge, MA, U.S.A.
[16] Goldsmith, J., 1976. “Autosegmental Phonology.” Thesis, MIT.
[17] Deng, L. and Sun, D.X., 1994. “A statistical approach to automatic speech recognition using the atomic speech units constructed from overlapping articulatory features”. J. Acoust. Soc. Am. Volume 95, Issue 5, pp. 2702-2719.
[18] Carson-Berndsen, J., 1998. “Time Map Phonology: Finite State Models and Event Logics in Speech Recognition”. Kluwer Academic Publishers, Dordrecht.
[19] King, S. and Taylor, P., 2000. “Detection of phonological features in continuous speech using neural networks.” Computer Speech & Language, Volume 14, Issue 4, pages 333 - 353.
[20] Kirchhoff, K., Fink, G. A. and Sagerer, G., 2002. “Combining acoustic and articulatory feature information for robust speech recognition.” Speech Communication. Volume 37, Issues 3-4, pages 303 - 319.
[21] Aioanei, D., Neugebauer, M. and Carson-Berndsen, J., 2005. “Efficient Phonetic Interpretation of Multilinear Feature Representations for Speech Recognition.” Language Technology Conference.
[22] Kane, M. and Carson-Berndsen, J., 2011 “Multiple source phoneme recognition aided by articulatory features.” In Proceedings of Springer-Verlag Lecture Notes in Computer Science (IEA/AIE)
[23] The-International-Phonetic-Alphabet-2005. *http://www.langsci.ucl.ac.uk/ipa/*.
[24] Scharenborg, O., Wan, V. and Moore, R. K., 2007. “Towards capturing fine phonetic variation in speech using articulatory features.” Speech Communication. Volume 49, Issues 10-11, Pages 811-826
[25] Chang, S., Wester, M. and Greenberg, S., 2005. “An elitist approach to automatic articulatory-acoustic feature classification for phonetic characterization of spoken language.” Speech Communication, Volume 47, Issue 3, Pages 290-311.
[26] Sakoe, H. and Chiba, S., 1978. “Dynamic programming algorithm optimization for spoken word recognition.” IEEE Transactions on Acoustics, Speech and Signal Processing, vol. 26, Issue 1, Pages 43-49.
[27] Garofolo, J., L. Lamel, W. Fisher, J. Fiscus, D. Pallett, and N. Dahlgren, 1993. The DARPA TIMIT Acoustic-Phonetic Continuous Speech Corpus CDROM.
[28] Steve, Y., Evermann, G., Gales, M., Hain, T., Kershaw, D., Liu, X., Moore, G. and Odell, J., Ollason, D., Povey, D., Valtchev, V. and Woodland, P., 2009. “Hidden Markov Model toolkit (HTK).” *http://htk.eng.cam.ac.uk/*, Version 3.4.1.

Automatic Identification of Arabic L2 Learners Origin

Mansour Alsulaiman, Bencherif Mohamed . A,
Ghulam Muhammad, Zulfiqar Ali,
Mohammed Al-Gabri
College of Computer and Information Sciences
King Saud University
Riyadh, Saudi Arabia
{msuliman, mbencherif, ghulam}@ksu.edu.sa

Ghassan H. Al Shatter, Saad A. Al-Kahtani,
Arabic Language Institute
King Saud University
Riyadh, Saudi Arabia
{galshatter,alkahtan}@ksu.edu.sa

Abstract— **The use of the computer for correcting pronunciation makes learning a second language (L2) more interesting. L2 learners will have instantaneous feedback for as long as they desire. Identifying L2 learners' first language allows choosing the suitable system for recognizing and correcting the pronunciation of the speaker. This paper presents the results of a system that classifies the speakers from three ethnic groups (Africans, Indonesians, and Pakistanis) according to their first language. It also presents the results of a system that detects whether or not the speaker is an Arab. The obtained results are encouraging. The classification system was able to classify the speakers into three ethnic origins with an average recognition rate of 93.33% (average of 100%, 90%, and 90%), while the ethnic detection system was able to detect if the speaker was an Arab or not with an average rate of 97.04%. We also investigated the Gaussian Mixture Models (GMM) with the Minimum Description Length (MDL) and showed that it was promising. An investigation was also conducted in order to determine the best digit (digits) that can indicate the ethnicity of the speaker.**

Keywords-component; Identification of first language; minimum description length; Auomatic pronounciation error detection; feature selection; Arabic Language.

I. INTRODUCTION

Challenges in acquiring a second language (L2) vary in their nature. Some are language related and others are learner related. Those related to language include but not limited to structural, developmental and phonetic. This study attempts to develop one part of a Computer Assisted Language Learning (CALL) program that is able to identify the background of learners of Arabic as second language (Arabic L2) based on their pronunciation of Arabic words. This attempt relies on the influence of learners' first language (L1) on pronouncing L2 sounds, in order to classify Arabic L2 learners into groups. Several studies found that L1 affect learners' of L2 pronunciation [1, 2, 3, 4]. Some believe that this effect is a negative influence which holds L2 learners from acquiring particular phonological skills [3]. An effect that, in turn, is influenced by the L2 learner age; the older the learners are the lesser they master the L2 pronunciation (Patkowski, 1990; Long, 1990; and Flege, Munro and MacKay, 1995 cited in [1]). However, learners from different L1 backgrounds show different reaction to L2 sounds. Pronunciation of L2 sounds depends in large part on the existence of that sound in the learner's L1 system. If the sound exists in both L1 and L2, then this is a case of positive transfer in which learners are expected to produce that sound with no trouble. If on the other hand, the sound in L2 does not exist in the learner's L1 sound system, then this is a case of negative transfer in which the learner of L2 will show difficulties in producing that sound [3].

Several studies found the phonological influence of L1 on L2 production useful in the application of Automatic Speech Recognition (ASR) technology to L2 learning [5, 6, 7]. These studies focused on the application of the technology to recognize learners' phonetic production, in order to identify speech type [7], learners' background [5], and pronunciation errors [8]. According to Doremalen [6], '[t]he application of … [ASR] technology to [L2] learning, and in particular to pronunciation training, has received growing attention in the last decade. However, one cannot use the same statement to describe the attention paid to ASR application to Arabic L2 learning. According to Olatunji [7], Arabic had received less attention than other languages. This in turn emphasizes the need for more empirical ASR studies on Arabic L2 learning. The present study provides one additional application of ASR technology to Arabic L2, and aims to classify Arabic L2 learners into groups based on their origin.

Arabic phonetic system includes number of sounds that might not exist in many other languages. for example, sounds such as /d̲/ (ض), /ħ/ (ح), /q/ (ق), /s̲/ (ص), /t̲/ (ط), /ʕ/ (ع), /χ/ (خ), and /ɣ/ (غ), do not exist in English, therefore it is expected that Arabic L2 learners of English background will find it difficult to pronounce these sounds. Khaldieh [8] stated that 'although there is one-to-one correspondence between graphemes and phonemes in Arabic, American readers of Arabic appear to have difficulty identifying words that have certain sounds. Nonetheless, not all Arabic L2 learners share the same difficulties in pronouncing those sounds. Depending on their background, learners may

pronounce some of the sounds above without difficulties, either because the sounds are available in the learner's L1 system or, in a special case of Arabic, the learners were trained to pronounce them for religious purposes (e.g. reading the holy Qur'an which is written and read in Arabic). For instance, Indonesian learners of Arabic L2 are expected to find troubles in producing the sounds /d̲/ (ض), /ħ/ (ح), /ʕ/ (ع), and /ɣ/ (غ). However, many of them will pronounce it correctly because they were trained to do so, in order to read religious texts. Other non-trained Indonesian learners may take longer time to pronounce these sounds. Pakistani Arabic L2 learners on the other hand, appear to have difficulty in pronouncing other sounds such as /d/ (د), and /t/ (ت), among others. This variability in the ability to pronounce Arabic sounds among Arabic L2 learners of different ethnicities, form a base for a framework that allows ASR designers to develop CALL program capable of identifying Arabic L2 learner's origin to a certain degree.

The CALL system that we developed uses Mel Frequency Cepstral Coefficients as the speech features. We may combine them with other features as in section 4. The modeling techniques of the system are Hidden Markov Models (HMM) or Gaussian Mixture Models (GMM). In this paper we propose an improvement to the way we apply GMM. The usual way to set the number of mixtures is to try many numbers of GMMs to find the number that shows the best performance. The improvement we suggest is to use minimum description length (MDL) to find the required number of mixtures. We denote the use of GMM and MDL by GMM-MDL. MDL will be explained in section II. In section III we describe the database of the speech. The experimental setup and the results are presented in section IV and V respectively. We conclude the paper in section VI.

II. Model Selection Using the Minimum Description Length and the Akaike Information Criteria

One critical issue in the estimation of the number of Gaussians in Multidimensional spaces is finding the appropriate number of clusters. One brute force method is to try as many clusters as possible, then select the model that gives the best data fitting, the stopping criteria can be the Akaike information criteria (AIC))[13], the Bayesian Information criteria (BIC), that rely on the log-likelihood of the data and the numbers of parameters of the model. The Matlab GMM built-in function uses the AIC as a stopping criterion and is defined as follows:

$$AIC = -2\,log\,(P(X|\Theta)) + 2 \times L \qquad [1]$$

Where:

- $log\,(P(X|\Theta))$: The log-likelihood of the Data X given the estimated model Θ, with diagonal covariance matrices.
- $L = M\,(1 + 2 * d)$
- M: The number of weighting components in the GMM
- d: Dimension of the MFCC features.

In this paper we propose the use of the Minimum Message Length (MDL) as a criterion for finding the best number of Gaussians. The use of MDL with GMM can be explained as follows: The GMM evaluation starts from a roughly heuristic huge number of Gaussians, then at each iteration of the expectation–maximization (EM) algorithm, the MDL criterion is computed, then near clusters are merged, decreasing the number of clusters, at the end one cluster remains. The MDL computed at each step is stored. The minimum MDL corresponds to the GMM model that best fits the data. MDL is defined as follows:

$$MDL = -log\big(P(X|\Theta)\big) + \frac{1}{2}L \times log(N) \qquad [2]$$

Where:

- N is the number of samples in the data X.

III. Database Description

The training and testing speech datasets used were from a rich database recorded in King Saud University [11]. This database consists of many selected words, sentences, long paragraphs, as well as questions. In addition to the richness in text, the database is rich in many other dimensions. It is rich in sessions (three sessions), in settings (two to three settings in each session), in recording environment (office, soundproof room, and cafeteria), in type of recording devices (Yamaha professional microphones, Yamaha mixer to low cost microphones), in nationalities (29 Nationalities), and in ethnicity (Arabs and Non-Arabs).

From the above database, we used the speech files for the following texts (Table 1 presents the alphabet distribution of the text):

- The digits “0” to “9” pronounced by 21 speakers from three ethnic groups (seven speakers per group), namely Indonesians (IDO), Pakistanis (PK), and Africans (AF).
- The digits “0” to “9” pronounced by 57Arab speakers and 57 Non-Arab speakers.
- 2 short sentences of 4 seconds long. These sentences have been used previously in SAAVB [12] due to their ability to capture the Saudi dialect.
- 2 long paragraphs of approximately 2 minutes duration.

Table 1: Alphabet distribution in selected speech subsets

	IPA	0	1	2	3	4	5	6	7	8	9	4,7,9	1,4,7,9	0,2,3,5,6,8	All digits	SAAVB sent.	Paragraphs
أ	ʔ		1	2	1	1				1		1	2	4	6	15	127
ب	b					1			1			2	2		2	3	17
ت	t				1	1	1	2	1	1	2	4	4	5	2	1	23
ث	θ			1	2					1				4	4	2	4
ج	ʒ															2	18
ح	ħ		1										1		1		15
خ	χ						1							1	1	1	4
د	d		1										1		1	1	18
ذ	ð															2	5
ر	ɾ	1		---		1						1	1	1	2	3	35
ز	z																4
س	s						1	1	1		1	2	2	2	4	1	18
ش	ʃ																5
ص	sˤ	1												1	1		5
ض	dˤ															1	6
ط	tˤ																6
ظ	ðˤ															1	4
ع	ʕ					1			1		1	3	3		3	1	21
غ	ɣ																8
ف	f	1												1	1	3	24
ق	q															1	11
ك	k																18
ل	l				1									1	1	10	71
م	m						1			1				2	2	1	41
ن	n			2						1				3	3		25
و	w															2	23
ه	h														1	1	38
ي	y									1				1	1	3	40

IV. The System Description

The system that we developed to classify and detect the ethnicity of the speaker is depicted in Figure 1. The text, the speech features, and the modeling technique depend on the investigation performed. For the classification we used HMM or GMM. For the detection, we used only GMM.

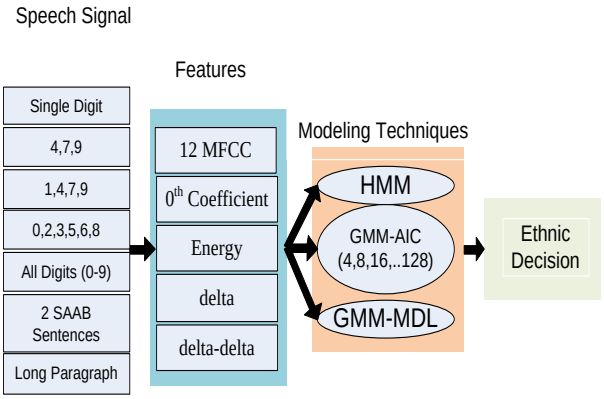


Figure 1: Block diagram of the system for Automatic Identification of Arabic L2 Learners Origin

V. Results

We conducted many investigations and performed many experiments to find the characteristics of the detection systems that will have the best performance. The characteristics include the modeling technique, the number of Gaussians for GMM, the number of states for HMM, the speech features, and the text of the speech. We also wanted to see which digit would produce better classification rate for each ethnic group. The classification investigation used the 21 speaker from the three ethnic groups while for the detection we used 57 Arabs (Saudis) and 57 non-Arabs. The speakers used in each experiment are divided into 70% for training and 30% for testing. The results of the investigations are presented in the next sections.

A. Ethnicity classification by HMM

The results of using HMM for Ethnicity classification in the initial experiments are given in figure 2, for a single random train/test distribution, while figure 3 presents the average results of 5 distinct random train/test distributions. Each distribution used 5 speakers for training and 2 speakers for testing. By comparing figures 2 and 3 we can see that the result of one run of any experiment is not enough to get a final conclusion. Hence in all the classification part of the paper the result we present is the average of 5 distinct random train/test distributions or datasets.

The first investigation tackled the digits (4, 7, and 9,) as a group, because each single digit contains the phoneme /ʕ/ (ع), which is known to be hardly pronounced by most of non-Arabs, hence, it might be the best choice to detect non-Arabs. The second combination added the digit ONE to the previous combination because it included the phoneme /ħ/ (ح) which is also hard to pronounce by non-Arabs. The third digit combination used the six remaining digits, then we used all digits and we also used the two SAAVB sentences. Figure 3 shows that the system had the highest accuracy in classifying the IDO. The figure also shows that the six-digits text was the optimal text though we anticipated that SAAVB sentences will be the optimal text.

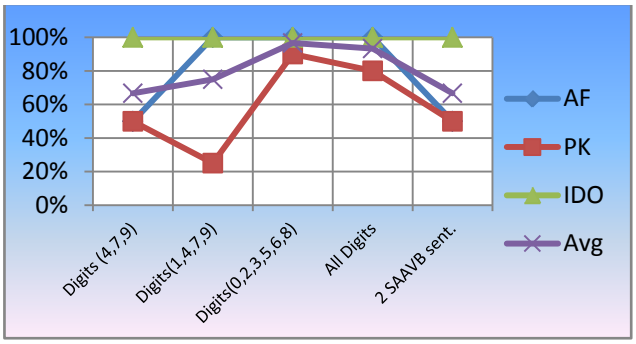


Figure 2: Ethnic Classification by HMM – initial investigation

The system parameters were: 12 MFCC for all experiments, 5 states with 2 Gaussians per state for the digits experiments, and 16 states with 4 Gaussians per state for the 2 sentences.

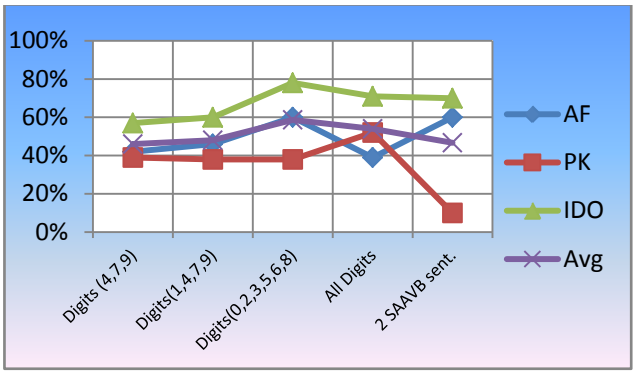


Figure 3: Ethnic Classification by HMM – Averaged over 5 repeated random train/test speech distributions

The numbers states for the model and the number of mixtures per state were selected as above because they gave the best results. Figure 4 is presented as an example of the effect of the number states on the classification results when the two SAAVB sentences are used. The average recognition rate was almost the same, but we can notice that the PK learners are still not relatively distinguished compared to the other ethnicities.

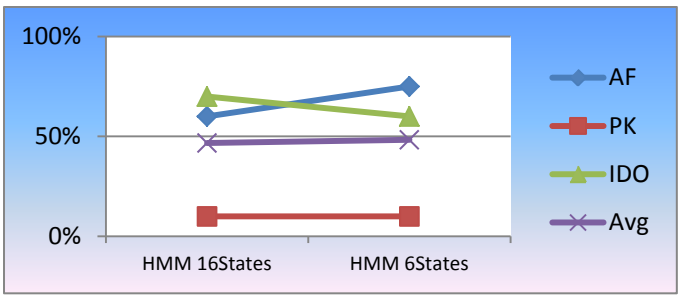


Figure 4: Effect of the number of HMM states on the results of the two sentences (4GMM per state)

B. *Ethnicity classification by GMM-MDL*

In the rest of experiments of the classification part we either used MFCC alone or appended by other features. We ran the experiments using all the speech features and reported only the result of the features that had the highest average recognition rate. The MFCC vectors are 12 coefficients. The MFCC can be appended by the 0^{th} coefficient ('0'), the energy ('e'), the first derivative ('d'), and the second derivative ('D'). The features, their corresponding abbreviation, and the number of their number of coefficients are shown in Table 2. We will use the features abbreviations in the figures or the discussions. In the figures, when no feature abbreviation is mentioned, it means only the 12 MFCC.

Table 2: Abbreviations of the features

Code	Abbreviation	Nbr. Coeff.
12MFCC		12
12MFCC+'0	'0'	13
12 MFCC+'0e'	'0e'	14
12 MFCC+'0ed'	'0ed'	28
12 MFCC+'0edD'	'0edD'	42
12MFCC+'dD'	'dD'	36

The results of using GMM-MDL for classifying the ethnic groups are given in figure 6 for the different speech subsets.

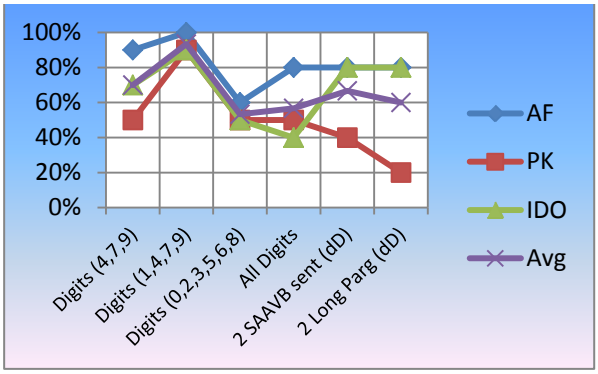


Figure 5: Ethnic classification using GMM-MDL

The GMM-MDL model presents a good separation scheme, which boosts the assumption of choosing the group (1, 4, 7, and 9) as a partition space where the recognition rate was 100%, 90%, and 90% for IDO, AF, and PK respectively. Hence the selected digits were able to let the system perform excellent classification. Similar to the HMM system, the GMM-MDL result of PK for the 2 SAAVB sentences was poor. The PK result when using the 2 paragraphs was very poor. This is not what we anticipated and need further research

C. *Finding the best Number of Mixtures for the GMM*

This investigation was done using the three digits (4, 7, and 9), all digits, the 2 SAAVB sentences, and the 2 paragraphs. The result using the GMM-MDL is also included for better decision making. The results are given in figures 6, 7, 8, and 9 respectively.

For the three digits (4,7,9), the classification rates using 12 MFCCs had better overall recognition rates, compared to the other features, hence, it is the only feature presented in figure 6. We can conclude that the effect of increasing the number of GMMs was not beneficial for IDO since the system had 100% rate in the low clustering space. For AF, increasing the number of mixtures result in increasing the rate. The PK was not captured by the system. The MDL did not show much improvement.

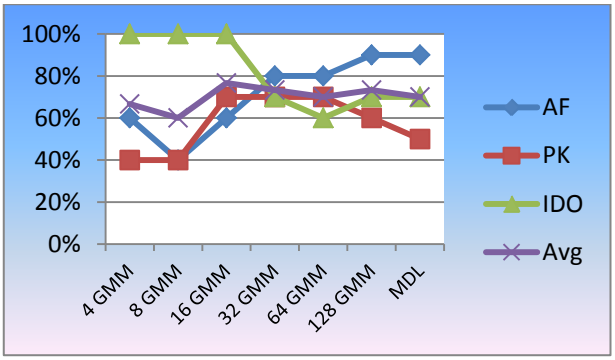


Figure 6: The effect of the number of GMMs on the three digits results

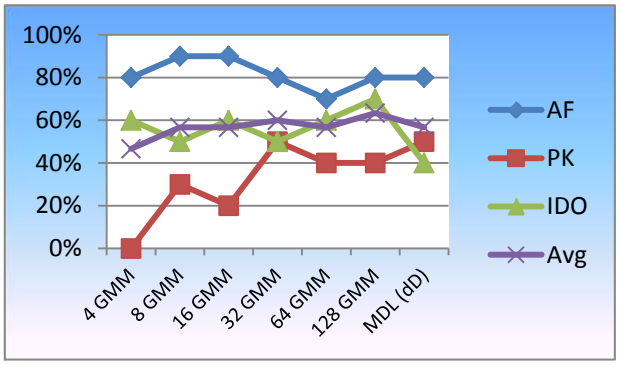


Figure 7: Effect of the number of GMMs on all digits results

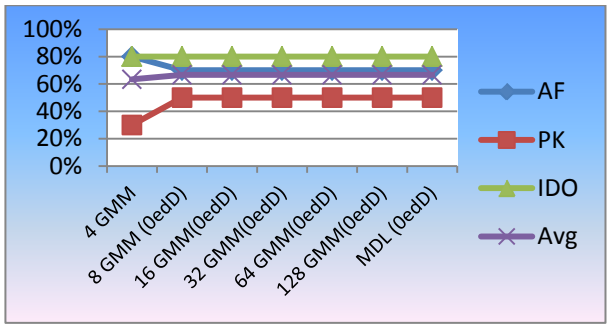


Figure 8: Effect of the number of Gaussians on the two sentences results

The two paragraph experiment uses long parameters, which have been included in the KSU speech database, for their ability to detect the pitch over a long session of speech, By using the 2 paragraphs, the AF presented better recognition rates, we can notice that the low clustering

dimension favors the AF while the high dimension favors the IDO, which is the inverse effect of figure 5.

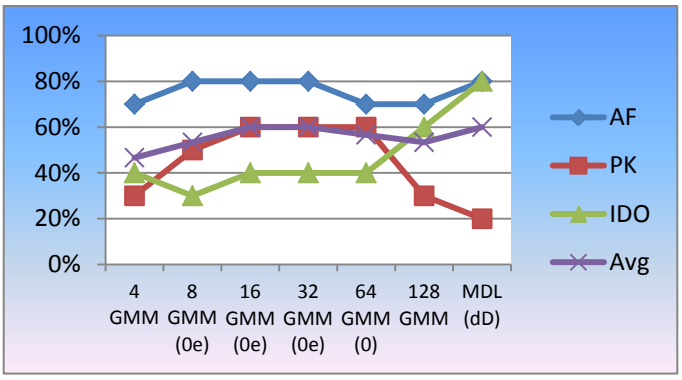


Figure 9: Effect of the number of GMMs on the 2 paragraphs results

D. Effect of Speech Features on Results of GMM-MDL

In figure 10, the results of GMM-MDL using the different speech features and the all digits text are presented.

It is noticed that the MFCC coefficients alone or with their first and second derivatives had the best recognition rates, and there is no need to use energies.

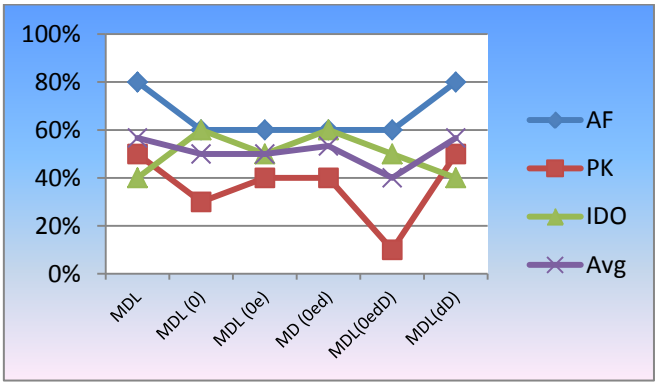


Figure 10: Ethnic Classification by GMM-MDL using all digits

For the 2 sentences, as presented in figure 11, the best result obtained for the IDO is 80% using 42 MFCCs (0edD), and for the AF is 80% using 36 MFCCs (dD). The PK presented a recognition rate of 70% using 13 MFCCs (0).

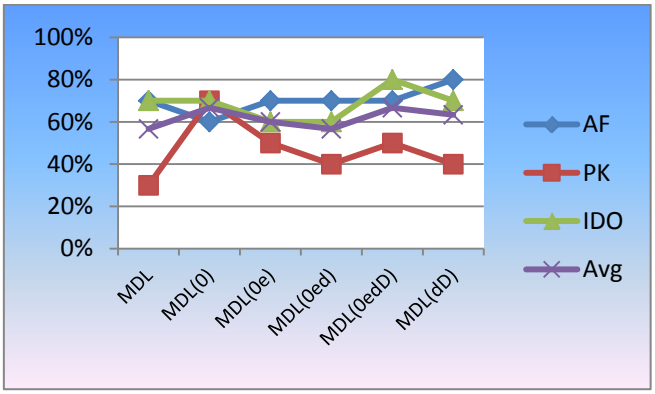


Figure 11: Ethnic Classification by GMM-MDL using the 2 sentences

E. Ethnic Classification by Single Digits

The goal of this section is to identify the digit, or the combination of digits that will have better performance in classifying the 3 ethnic groups. We do this for the system that uses GMM and the system that uses HMM. For the GMM system figure 12, presents the recognition rate per digit. Digit '3' distinguishes the AF (100%), digit '4' distinguishes the IDO (100%), and the PK did not have any distinguishing digit

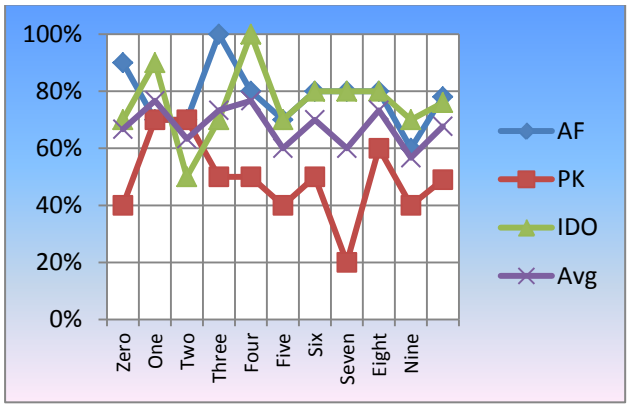


Figure 12: Ethnic Classification by Single Digits using GMM

For the HMM system, with one HMM model per digit, it can be remarked in figure 13, that the digit '0' distinguishes the AF(90%), the digit '5' distinguishes the IDO (95%), and the PK did not have any distinguishing digit.

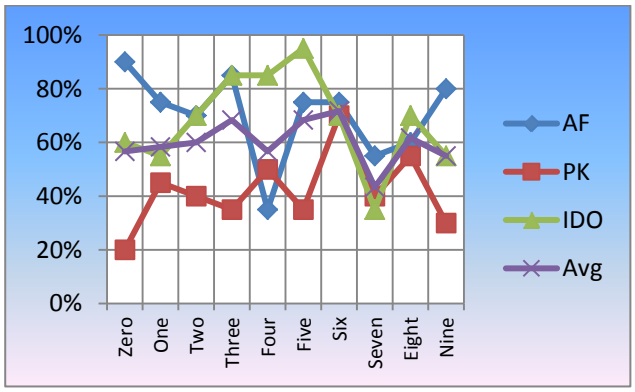


Figure 13: Ethnic Classification by Single Digits using HMM

F. Ethnicity Detection

Ethnicity detection system determines if a given speaker is Arab or non-Arab. Different experiments were performed by using the three digits (4, 7, and 9) and the two SAAVB sentences pronounced by 114 speakers (57 Arab (AR), and 57 Non-Arab (NAR)). All Arabs are Saudis, while non-Arabs originated from 5 ethnic groups from 11 nationalities. Forty speakers are used for system training and the remaining 17 are used for the testing.

A recognition rate of 97.05% (100% for AR and 94.12% for NAR) is achieved with 36 MFCC and 96 Gaussians for the 2 SAAVB sentences. The results are depicted in figure 14. The maximum detection rate obtained for the three digits

is 82.35% with 36 MFCC and 128 Gaussians; these results are provided in figure 15. Overall, results with 36 MFCC are better than 12 and 24 MFCC.

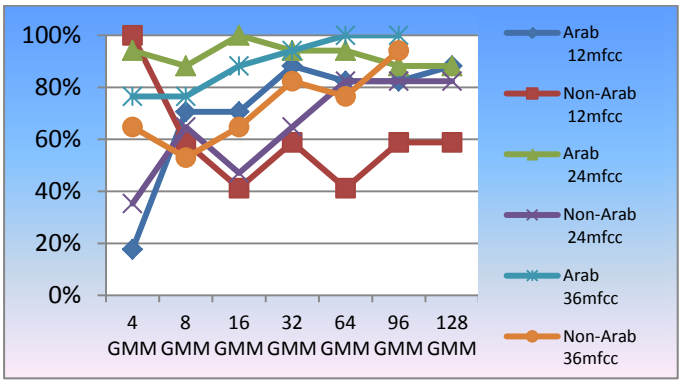


Figure 14: Ethnicity detection using the 2sentences

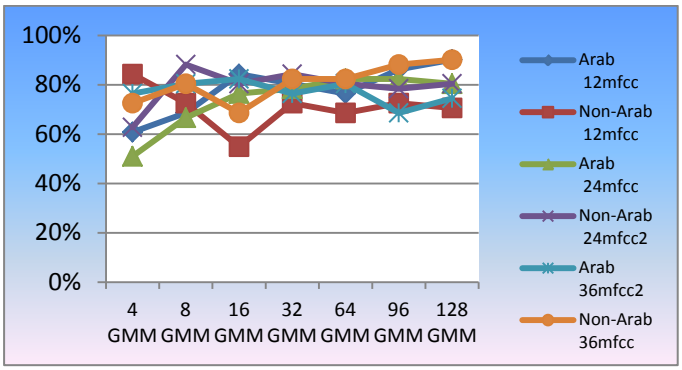


Figure 15: Ethnicity detection using the (4,7,9) digits

VI. Conclusion

The classification by using HMM was very good to excellent for IDO, medium for AF, and poor for PK. Compared to HMM, the GMM-MDL gave excellent results for both IDO and AF (100% at three different texts), but the result of PK was not as good. In fact the result of PK was poor in most experiments, this needs further investigation.

We also found that the four digits (1, 4, 7, and 9) gave an excellent classification rates: 100%, 90%, and 90% for IDO, AF, and PAK respectively. So even for the PAK it gave excellent result. The 2 sentences and the 2 paragraphs had more phonemes (in fact the 2 paragraphs contain all the phonemes with some repetitions), but their performance was clearly unsatisfactory. This is very interesting and need further study. Thus, by carefully selecting the text, high recognition rates can be reached. This was also confirmed by the single digits investigation.

By comparing the results of GMM and GMM-MDL, it is seen that GMM-MDL was better in many experiments. This shows that GMM-MDL is promising and needs more investigation.

In the detection part, an excellent identification rate of 97.03% with 36 MFCC and 96 Gaussians was obtained.

This initial investigation shows that by careful design of the system and selecting the suitable text, we can get excellent classification and identification rates. We will investigate this in future research using larger speech database.

Acknowledgment

This work is supported by the National Plan for Science and Technology in King Saud University under grant number 08-INF167-02. The authors are grateful for this support

References

[1] Flege, James E. and Elaina M. Frieda, Amount of Native-Language (L1) Use Affects the Pronunciation of an L2, Journal of Phonetics, 25, pp: 169- 186,1997.

[2] Guion, S., Flege, James E., Akahane-Yamada, Reiko and Pruitt, JC " The Effect of L1 Use on Pronunciation in Quichua-Spanish Bilinguals", Journal of Phonetics, 28, pp: 27-42,2000.

[3] Keys, K. (2002), First Language Information on the spoken English of Brazilian Students of EFL, ELT Journal, 56/1, pp: 41-46

[4] McAllister, R. Flege, J., and Piske, T., The Influence of L1 on the Acquisition of Swedish Quantity by Native Speakers of Spanish, English and Estonian, Journal of Phonetics, 30, pp: 229-258,2002.

[5] Bouselmi, D. Fohr, I. Illina, and J.-P. Haton, “Discriminative Phoneme Sequences Extraction for Non-Native Speaker’s Origin Classification”, in ISSPA, 2007.

[6] Doremalen, J. et al (2009), Automatic Detection of Vowel Pronunciation Errors Using Multiple Information sources, IEEE, pp: 580-585.

[7] Olatunji, S. et al (2010), I dentification of Question and Non-Question Segments in Arabic Monology Based on Prosodic Features Using Type-2 Fuzzy Logic System, Second International conference on Computational Intelligence, Modelling and Simulation, IEEE computer Society, pp: 149-153.

[8] Khaldieh, S (1996), Word Recognition of Arabic as a Foreign Langauge by American Learners: The Role of Phonology and Script, Journal of American Arabic Teachers Association, pp: 129-151, USA.

[9] Nannen, "The Paradox of Overfitting", Master’s thesis, Rijksuniversiteit Groningen, the Netherlands, April 2003.

[10] F. Pernkopf, D. Bouchaffra, " Genetic-Based EM Algorithm for Learning Gaussian Mixture Models ", IEEE transaction on pattern analysis and machine intelligence, Vol 27, No8, 2005.

[11] Awais Mahmood, Mohamed A. Bencherif ,Mansour M. Alsulaiman, and Ghulam Muhammad," Verification of a Rich Arabic Speech Database", Proceedings of the International Conference on Speech Database and Assessments(COCOSDA 2011), HsinChu,Taiwan.pp 100-105.

[12] Saudi Accented Arabic Voice Bank, Mansour Alghamdi, Fayez Alhargan, Mohammed Alkanhal, Ashraf Alkhairy, Munir Eldesouki and Ammar Alenazi, J. King Saud University, Vol. 20, Comp. & Info. Sci., pp. 45-62.

[13] M. F. Abu El-Yazeed , M. A. El Gamal, M. M. H. El Ayadi, "On the Determination of Optimal Model Order for GMM-Based Text-Independent Speaker Identification", EURASIP, Journal on Applied Signal Processing, 2004:8, 1078–1087, Hindawi Publishing Corporation.

Using the Wizard-of-Oz Framework in a Pronunciation Training System for Providing User Feedback and Instructions

João P. Cabral*, Mark Kane*, Zeeshan Ahmed*, Éva Székely*, Amalia Zahra*,
Kalu U. Ogbureke*, Peter Cahill*, Julie Carson-Berndsen* and Stephan Schlögl†
*School of Computer Science and Informatics, University College Dublin, Ireland
Email: see http://muster.ucd.ie
†School of Computer Science and Statistics, Trinity College Dublin, Ireland
Email: schlogls@tcd.ie

***Index Terms*—DEMO, MySpeech, Wizard-of-Oz, Pronunciation training**

I. Introduction

A prototype of a computer-assisted pronunciation training system called MySpeech is showcased in this demo. The web-based interface of the MySpeech system enables users to select a sentence from different domains, such as greetings, the difficulty level and contains both recording and playback functionalities. The interface is also used to provide instructions and feedback messages based on the pronunciation errors detected in their recorded speech by the system.

MySpeech uses an automatic speech recognition (ASR) method for detecting mispronunciation in the speech recorded by the user which is similar to [1]. However, this method was adapted to introduce difficulty levels in the pronunciation training of MySpeech, as proposed in [2] and clearly indicates that broad phonetic/phonological groups are suitable for tackling mispronunciations by non-native speakers. Recent developments include a spoken term detection front-end that explicitly uses underspecification to tackle pronunciation variation resulting in an increase in performance [3].

Both the pronunciation analysis component and the web interface are connected to a database. The database contains the audio and text data for the pronunciation practice exercise. It is also used to store data obtained from the interaction of each student with the system. The aim of collecting the user's data is to build a personalised student model that can be used to adapt the system to the user and to develop a pedagogical model. For example, the analysis of the pronunciation errors stored for a student could be used to automatically predict the appropriate difficulty level for that student. It could also be used to detect the most frequent types of pronunciation errors, in order to automatically suggest words containing those sounds for the student to practice. Other types of student data could also be used for adaptation, such as the recorded speech to adapt the acoustic models of ASR to the speaker.

One current limitation of the MySpeech system is that feedback and instructions given to a user are not automatically generated. The WebWOZ Wizard-of-Oz platform (http://www.webwoz.com) was integrated into the MySpeech system, in order to enable a human (who acts as a wizard) to give feedback and instructions to the practising user, while the user is not aware that there is another person involved in the communication. The Wizard-of-Oz (WOZ) method has been used before in language learning applications. For example, it was used to study a dialogue strategy in [4]. It was also employed to test and refine the human-computer interface and feedback display of a computer-based speech training aid called ARTUR [5]. In this demo, WOZ is used in a different context, namely to enable a semi-automatic operation of the MySpeech system, in which the wizard has access to the pronunciation analysis results computed by the system and provides feedback to the user based on those results. Another function of the wizard is to guide the student through the selection of sentences and control the progression of the student through their skills. For example, a student starts at the "easy" level and after practicing for some time at this level she is asked (by the Wizard) to progress to the next level. The data collected from the wizard will also be used to further improve the system.

II. Demonstration

A. Web Interface

Participants in the demo will use the MySpeech system to train their English pronunciation. Figure 1 shows a screenshot of the MySpeech web interface, which consists of several numbered panels. In panel 1 the user can select the language. The second panel allows the user to adapt the difficulty level ("easy", "medium", or "hard"). Next, there is a category panel (panel 3), so that for example, the category "greetings" can be associated with several phrases related to this domain. The different sentences are then chosen in panel 4. The audio players embedded in the interface are used by the users to listen to the selected sentence spoken by a native speaker (panel 5) and to record their own version of the same sentence and consequently submit it to the system (panel 6). The user can then submit the recording for the system to evaluate the

pronunciation. In the full automatic operation mode (without using the WOZ platform), the feedback panel (panel 7) shows the detected mispronunciation errors of a submitted utterance using darker colours. In this example, the submitted utterance corresponds to the sentence: *See you in the morning*. The other operation mode which requires the wizard interaction is explained in the next section.

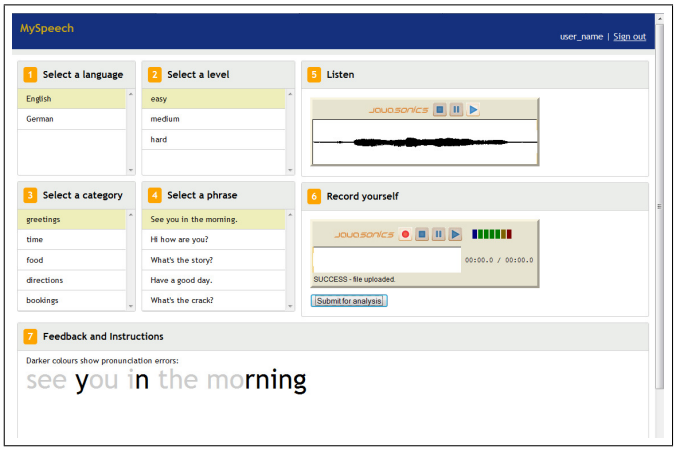


Fig. 1. Screenshot of the MySpeech web interface.

B. The Wizard-of-Oz Setup

A voice-over-IP system is used to give the wizard a real-time visualisation of the user's screen, and to transmit everything a user is saying (the user is not aware of this transmission). Furthermore, the wizard has access to the pronunciation analysis results computed by the system (displayed on a second screen). The wizard's task is to interpret a result and consequently to transform it into an appropriate textual feedback to be sent to the user. A screenshot of the WOZ interface is shown in Figure 2. The interface allows for selecting predefined sentences for instructions as well as feedback. Choosing from predefined sentences as opposed to typing feedback in real-time allows for a quicker response.

To decrease the time a wizard searches for an appropriate response, sentences are grouped into different panels. For example, the "difficulty" panel contains sentences that prompt the user to switch to a different difficulty level, whereas the "phrases" panel contains sentences that prompt her to select a different phrase. The panel called "feedback" contains sentences for indicating where the individual mispronunciations errors are within the sentence or word. There is also a panel with encouragement messages (called "positive"), which offers a way of motivating the user and indicating that pronunciation assessment was positive. Table I shows examples of the corrective and positive feedback sentences. Finally, the panel called "free text" allows a wizard to input any text, or edit an already predefined sentence from one of the other panels, before sending it to the user.

During the demo, one of the authors assumes the role of wizard and was trained beforehand to follow a simple

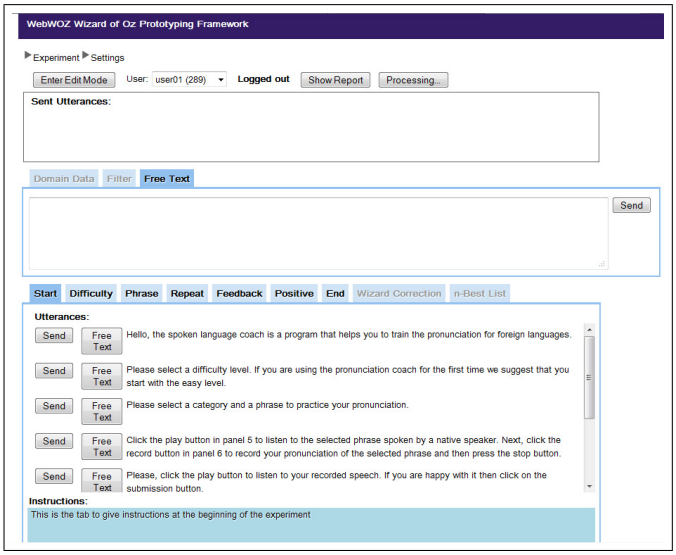


Fig. 2. Screenshot of the WOZ web interface.

interaction model to guide the user through the learning exercise. However, wizard has the freedom to alter the model as she wishes (exploring the interaction space).

Corrective feedback messages
You mispronounced the last part of the word
Please try to emphasize
You mispronounced the word
Positive feedback messages
Perfect, you pronounced the phrase correctly
You are showing some improvement
You are almost there

TABLE I
EXAMPLES OF MESSAGES FROM THE "FEEDBACK" AND "POSITIVE" PANELS OF THE WOZ INTERFACE.

ACKNOWLEDGMENT

This research is supported by the Science Foundation Ireland (Grant 07 / CE / I 1142) as part of the Centre for Next Generation Localisation (www.cngl.ie).

REFERENCES

[1] Witt, S. M. and Young, S. J., "Phone-level pronunciation scoring and assessment for interactive language learning", Speech Communication, Vol. 30, pp. 95–108, 2000.
[2] Kane, M., Cabral, J. P., Zahra, A., Carson-Berndsen, J., "Introducing Difficulty-Levels in Pronunciation Learning" Proc. of SLaTE, Italy, 2011.
[3] Kane, M., Ahmed, Z. and Carson-Berndsen, J., "Underspecification in Pronunciation Variation", In Proc. of International Symposium on Automatic Detection of Errors in Pronunciation Training (IS ADEPT), Stockholm, 2012.
[4] Ehsani, F., Bernstein, J., Najmi, A.,"An interactive dialog system for learning Japanese", Speech Communication, 30(2-3), pp. 167–178, 2000.
[5] Bälter, O., Engwall, O., Öster, A., and Kjellström, H., "Wizard-of-Oz test of ARTUR: a computer-based speech training system with articulation correction", Proc. of ASSETS, pp. 36–43, 2005.

Comparing sound inventories for CAPT

Jacques Koreman, Olaf Husby
Department of Language and Communication Studies
Norwegian University of Science and Technology (NTNU)
Trondheim, Norway
jacques.koreman@ntnu.no, olaf.husby@ntnu.no

Preben Wik
Department of Speech, Music and Hearing
Royal Institute of Technology (KTH)
Stockholm, Sweden
preben@speech.kth.se

***Abstract*—This demo introduces L1-L2*map*, a tool for contrastive analysis of the sound inventories of source and target language pairs. The demo will give opportunity to discuss its use and limitations, as well as its potential for further development.**

Contrastive analysis, multi-lingual, CAPT, demo

I. L1-L2*MAP*

The multi-lingual tool L1-L2*map* allows users (especially CAPT developers) to compare the phonemic sound inventories of source and target language pairs for foreign language learning. The tool has been developed on the basis of the UPSID database [1], and has now been extended to over 500 languages. It is implemented as a wiki [2] and is available free of charge. It can be integrated into any CAPT system by sending a simple query to the server running L1-L2*map*. The query results in a list of sounds in the target language that are not part of the specific learner's source language and which may potentially cause pronunciation problems.

Although this is not necessary for CAPT developers, the results can also be visualized in charts with a similar lay-out to the IPA consonant and vowel charts [3]. Figs. 1a and 1b show a contrastive analysis of Mandarin Chinese compared with south-eastern Norwegian. The colours in L1-L2*map* have intuitively interpretable functions: green indicates sounds that L1 and L2 have in common; blue indicates sounds that are only used in the learner's native language (L1, cf. blue language box at the top of the figures); and red indicates sounds that only occur in the target language (L2, cf. red language box). It is the list of sounds which are indicated on a red background which a query returns, so that the CAPT developer can direct the user to corresponding exercises for these unfamiliar sounds.

Figure 1a. Contrastive analysis comparing the consonant inventories of Mandarin Chinese and (south-eastern) Norwegian

II. EMBEDDING IN CAPT

The tool is used in a CAPT system for Norwegian [4] which is based on the VILLE system developed for Swedish [5]. The CAPT system consists of two parts [6]. The first part is devoted to vocabulary training, and exposes learners to different dialects of Norwegian (which has no accepted pronunciation standard) through listening, pronunciation and writing exercises for "1000 words and expressions". The second part consists of minimal pair/set listening exercises. In this part of the system, L1-L2*map* is used to select relevant exercises for each individual learner of Norwegian depending on his/her native language.

III. SYLLABLE POSITION

The use of consonants in different syllable positions can vary across languages. Because language learners often have problems with pronouncing known sounds in their native language when they appear in "unusual" syllable positions, information about the use of consonants in onset, nucleus and coda can also be provided in L1-L2*map*. This information is important in CAPT systems, for example to help German learners of Norwegian (or English) to learn to use lenis (voiced) consonants in syllable-final positions, where they often realize them as fortis (voiceless) sounds. Another example is the strong restrictions on the use of consonants in Mandarin Chinese, where only /n/ and /N/ occur syllable-finally, while Norwegian (and English) allow a much wider range of consonants in that position [7]. Though familiar with many of the sounds, learners need to practise pronouncing them in unusual syllable positions. In cases where both language that are compared are defined for the occurrence of sounds in different syllable positions (onset, nucleus, coda), this information is included in the query result from L1-L2*map* and can be used to direct the learner to minimal pair/set exercises for unfamiliar sounds in specific syllable positions.

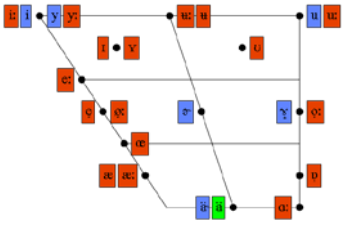


Figure 1b. Contrastive analysis comparing the vowel inventories of Mandarin Chinese and (south-eastern) Norwegian

IV. LIMITATIONS

It is well known that a contrastive analysis cannot predict all the mistakes learners make (cf. references in [6]), and particularly it does not predict what substitutions learners make [8]. For example, learners of English make different substitutions for dental fricatives depending on the language background [9]. These substitutions would provide an optimal basis for the selection of minimal pair exercises. Automatic error detection can be a solution to this, but no ASR or signal processing techniques are available for Norwegian CAPT. Such techniques also do not yet take into account the complex systematic variation in native speakers' realization of speech sounds in talk-in-interaction, which is far outside the scope of present-day approaches [10]. Our aim is to collect data about actual substitutions by foreign learners of Norwegian, and we have started making sound recordings in a pilot project for five languages spoken by large groups of immigrants to Norway.

ACKNOWLEDGMENT

We thank Øyvind Bech for implementation of L1-L2*map* and for useful contributions to discussion.

REFERENCES

[1] I. Maddieson, Patterns of Sounds. Cambridge: Cambridge University Press, 1984.

[2] Ø. Bech, J. Koreman, O. Husby, P. Wik, L1-L2*map*, http://calst.hf.ntnu.no/L1-L2map, 2011.

[3] International Phonetic Association, Handbook of the International Phonetic Association. Cambridge: Cambridge University Press, 1999.

[4] J. Koreman, O. Husby, P. Wik, Å. Øvregaard, E. Albertsen, S. Nefzaoui, E. Skarpnes, "CALST - computer-assisted listening and speaking tutor," http://www.ntnu.edu/isk/calst, 2011.

[5] P. Wik and A. Hjalmarsson, "Embodied conversational agents in computer assisted language learning," Speech Communication 51(10), 1024-1037, 2009.

[6] Husby, O., & Øvregaard, Å., Wik, P., Bech, Ø., Albertsen, E., Nefzaoui, S., Skarpnes, E. & Koreman, J. (2011). Dealing with L1 background and L2 dialects in Norwegian CAPT, Proc. of the workshop on Speech and Language Technology in Education (SLaTE2011), Venice (Italy).

[7] J. Koreman, Ø. Bech, O. Husby and P. Wik, "L1-L2map: a tool for multi-lingual contrastive analysis," Proc. ICPhS, 2011.

[8] S. Weinberger, "Speech accent archive," http://accent.gmu.edu, 2012.

[9] L. Lombardi, "Second language data and constraints on Manner: explaining substitutions for the English interdentals," Second Language Research 19(3), 225–250, 2003.

[10] J. Local, "Variable domains and variable relevance: interpreting phonetic exponents," J. Phon. 31, 321-339, 2003.

The Danish Simulator

Learning language and culture through gaming

Thomas K. Hansen
Resource Center for Integration
www.vifin.dk / www.dansksimulatoren.dk
Vejle, Denmark
thkha@vejle.dk

Abstract— **In the year 965 A. D. The Danish King, Harald Bluetooth, erected a monument - a stone, signifying the end of the Viking era and the conversion of the Danes to Christianity.**

You are: Bob Johnson, a journalist from Hawaii.

Your mission: find the stone.

Keywords; Language, Culture, Speech recognition; gaming; pronunciation;

I. THE DANISH SIMULATOR

The Danish Simulator (DS) is an online, browser-based language- and culture learning platform for Danish. It contains interactive lessons for learning communicative skills and a realistic 3D virtual representation of the area around Vejle city in Denmark, which the learner has to navigate, while using the acquired communicative and cultural skills.

II. THE PRONUNCIATION TRAINER

Part of the DS is the Pronunciation Trainer (PT). The goal of the PT is to detect and correct single-segment pronunciation errors in foreign learners. This is currently done by an algorithm which cross-references the results of multiple mis-pronounced words with a database of known common errors.

The pronunciation error is identified and the learner is given the option of switching from regular pronunciation exercises to exercises involving a particularly difficult sound.

We are currently working on enabling the system to provide more specific feedback to en encountered problem, via animations of correct pronunciation of a detected error.

I. THE PRONUNCIATION TRAINER – FUNCTIONALITY

The PT consists of approximately 500 phonetically balanced and rich words which are statistically 'commonly' occurring in Danish. The learner can choose words arranged on an easy to difficult scale in terms of pronunciation. The learner can also choose certain categories of words and can furthermore search for sound-combinations which are deemed hard to pronounce.

Figure 1 displays the user interface. The learner has tried to pronounce the word 'arbejde' (work) but has not been recognized, which is signaled by King Harald through the thumbs down gesture. A number of words have been mispronounced so far, prompting the PT to signal that the learner may have a problem with the sound <e> as shown in the infobox below the image of King Harald. The learner can now go to a practice section containing all the words with the problem sound.

Figure 1: The PT interface

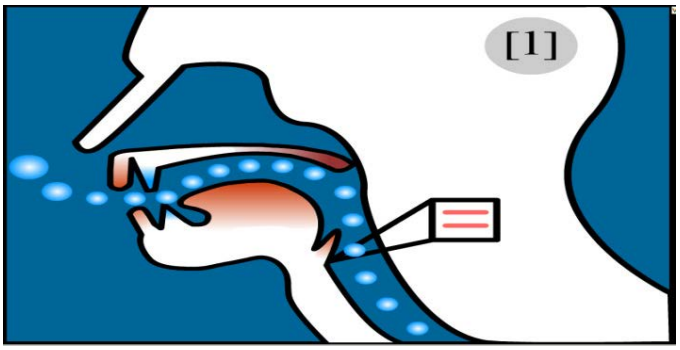


Figure 2: Feedback 'to be' in the PT

Figure 2 shows a way of providing the learner with more specific feedback on how to remedy pronunciation errors is via small animations of how to pronounce a particular sound as also done in the Vifin product called www.dansk.nu

Keynote speaker demos

I. Florian Hönig, University of Erlangen
II. Gary Pelton, CarnegieSpeech
III. Bryan Pellom, RosettaStone
IV. Lewis Johnson, Alelo

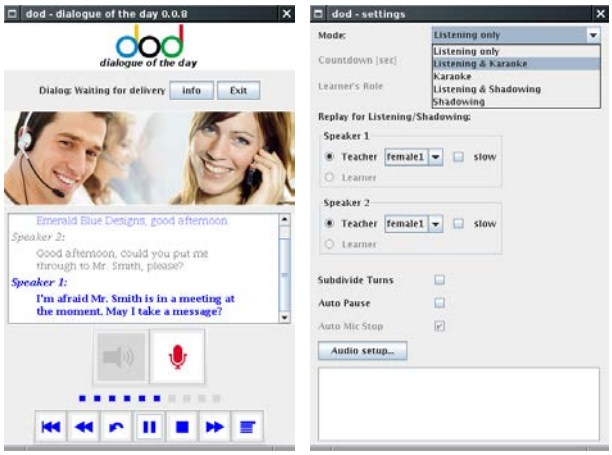
***Abstract*— Four of the invited speakers will demonstrate systems related to their keynote papers included earlier in these proceedings. This description summarizes their demos; refer to the respective keynote papers for more details. The author numbers above refer to the corresponding demo.**

I. DIALOGUE OF THE DAY

We (Florian Hönig, Anton Batliner and Elmar Nöth) present a web-based software prototype for practising prosody in pre-scripted dialogues and for training pronunciation with an emphasis on prosody, developed within the German research project AUWL (Grant No. KF2027104ED0). Using it, the learner can

a) rehearse realistic communication tasks, and
b) train his or her pronunciation skills using more natural speech than when e.g. just reading off isolated prompts from the screen.

The learner enacts the dialogue with a reference speaker as a dialogue partner. In doing so, he can either have his lines prompted by a reference speaker and repeat afterwards, or directly read them off the screen (karaoke), or speak simultaneously with a reference speaker (shadowing). Taking into account less proficient learners, one can choose between reference recordings spoken in a normal or in a slow tempo, and longer dialogue steps can be subdivided. Options for choosing from different reference speakers, swapping roles, restarting from an arbitrary position, replaying the latest own version of a dialogue step, replaying the whole enacted dialogue, or using own recordings for the dialogue partner, complete the versatile training tool. Global and local feedback is given with respect to the prosodic quality of the learner's productions (see the corresponding keynote paper in these proceedings for technical details).



II. CARNEGIESPEECH - NATIVE ACCENT

NativeAccent delivers assessment and instruction in pronunciation, word and sentence stress, fluency and spoken English grammar. The program dynamically organizes and presents curriculum that is customized for each student, based on gender, language of origin, and individual skill levels based on initial and medial assessments of these skills.

NativeAccent speech recognition evaluates the students' spoken English measured against multiple male or female American English voice models in a process patented by Carnegie Speech.

The Intelligent Tutor software guides students through the curriculum to rapidly improve their comprehensibility in English. Intelligent tutoring has been shown to accelerate learning by a factor of 3.

Case studies and research show that students who use NativeAccent can achieve significant (238%) increases in comprehensibility in English in 10 or more hours of instruction. The mean time to best improvement is about 30 hours of instruction (refer to www.edurep.com/highered for details).

III. ROSETTASTONE - REFLEX / TOTALE

The RosettaStone products ReFLEX and/or TOTALe will be introduced and demonstrated. The online training program ReFLEX, which combines games and other activities that practice sound skills, simulated conversational narratives that rely on speech recognition, and one-on-one live human interaction, is described in Bryan Pellom's keynote paper in these proceedings.

IV. ALELO - LANGUAGE AND CULTURAL TRAINING

Alelo's language and culture products to help learners develop communicative competence, described in detail in Lewis Johnson's keynote paper in these proceedings, will be demonstrated.

Author Index

ISBN 978-91-7501-402-9