A COMPARISON-BASED APPROACH TO MISPRONUNCIATION DETECTION

Ann Lee, James Glass

MIT Computer Science and Artificial Intelligence Laboratory,
Cambridge, Massachusetts 02139, USA
{annlee, glass}@mit.edu

ABSTRACT

The task of mispronunciation detection for language learning is typically accomplished via automatic speech recognition (ASR). Unfortunately, less than 2% of the world's languages have an ASR capability, and the conventional process of creating an ASR system requires large quantities of expensive, annotated data. In this paper we report on our efforts to develop a comparison-based framework for detecting word-level mispronunciations in nonnative speech. Dynamic time warping (DTW) is carried out between a student's (nonnative speaker) utterance and a teacher's (native speaker) utterance, and we focus on extracting word-level and phone-level features that describe the degree of mis-alignment in the warping path and the distance matrix. Experimental results on a Chinese University of Hong Kong (CUHK) nonnative corpus show that the proposed framework improves the relative performance on a mispronounced word detection task by nearly 50% compared to an approach that only considers DTW alignment scores.

Index Terms— language learning, mispronunciation detection, dynamic time warping

1. INTRODUCTION

Computer-Aided Language Learning (CALL) systems have gained popularity due to the flexibility they provide to empower students to practice their language skills at their own pace. A more specific CALL sub-area called Computer-Aided Pronunciation Training (CAPT) focuses on topics such as detecting mispronunciation in nonnative speech.

Automatic speech recognition (ASR) technology is a natural component of both CALL and CAPT systems, and there has been considerable ASR research in both of these areas. However, conventional ASR technology is language specific, and the process of training a recognizer for a new language typically requires extensive (and expensive) human efforts to record and annotate the necessary training data. While ASR technology can be used for students learning English or Mandarin, such practices become much more problematic for students trying to learn a rare language. To put this issue in a more global context, there are estimates of around 7,000 languages in the world [1], among which 330 languages are with more than a million speakers, while language-specific ASR technology is available for approximately 80 languages [2]. Given these estimates, it is reasonable to say that over 98% of the world's languages do not have ASR capability. While popular languages receive much of the attention and financial resources, we seek to explore how speech technology can help in situations where less financial support for developing conventional ASR capability is available.

In this paper, a comparison-based mispronunciation detection framework is proposed and evaluated. The approach is inspired by the previous success in applying posteriorgram-based features to the task of unsupervised keyword spotting [3, 4], which is essentially a comparison task. In our framework, a student's utterance is directly compared with a teacher's through dynamic time warping (DTW). The assumption is that the student reads the given scripts and that for every script in the teaching material, there is at least one recording from a native speaker of the target language, and we have word-level timing information of the recording for the native speaker. Although this is a relatively narrow CALL application, it is quite reasonable for students to practice their initial speaking skills this way, and it would not be difficult to obtain spoken examples of read speech from native speakers. With these components, we seek to detect word-level mispronunciation by locating poorly matching alignment regions based on features extracted from either conventional spectral or posteriorgram representations.

The remainder of the paper is organized as follows. After introducing background and related work in the next section, we discuss in detail the two main components: word segmentation and mispronunciation detection. Following this, we present experimental results and suggest future work based on our findings.

2. BACKGROUND AND RELATED WORK

This section reviews previous work on individual pronunciation error detection and pattern matching techniques, which motivate the core design of our framework.

2.1. Pinpoint Pronunciation Error Detection

ASR technology can be applied to CAPT in many different ways. Kewley-Port et. al [5] used an isolated-word, template-

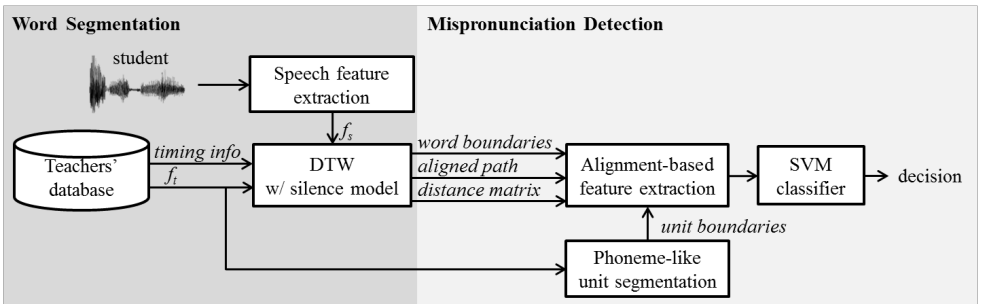


Fig. 1: System overview (with single reference speaker)

based recognizer which coded the spectrum of an input utterance into a series of 16-bit binary vectors, and compared that to the stored templates by computing the percentage of matching bits, which was then used as the score for articulation.

HMM-based log-likelihood scores and log-posterior probability scores have been extensively used for mispronunciation detection. Witt and Young [6] proposed a goodness of pronunciation (GOP) score based on the log-likelihood scores normalized by the duration of each phone segment. In this model, phone dependent thresholds are set to judge whether each phone from the forced alignment is mispronounced or not. Franco et. al [7] trained three recognizers by using data of different levels of nativeness and considered the ratio of the log-posterior probability-based scores.

Other people have focused on extracting useful information from the speech signal. Strik et. al [8] explored the use of acoustic phonetic features, such as log root-mean-square energy and zero-crossing rate, for distinguishing velar fricative and velar plosive. Minematsu et. al [9] proposed an acoustic universal structure in speech which excludes non-linguistic information.

Some approaches incorporate the knowledge of the students' native language into consideration. Meng et. al [10] incorporated possible phonetic confusions which were predicted by systematically comparing phonology between English and Cantonese into a lexicon for speech recognition. Harrison et. al [11] considered context-sensitive phonological rules rather than context-insensitive rules. Most recently, Wang and Lee [12] further integrated GOP scores with error pattern detectors to improve the performance on detecting mispronunciation within a group of students from 36 different countries learning Mandarin Chinese.

2.2. Posteriorgram-based Pattern Matching

Recently, posterior features with dynamic time warping (DTW) alignment have been successfully applied to the facilitation of unsupervised spoken keyword detection [3, 4]. A posteriorgram is a vector of posterior probabilities over some predefined classes. It can be viewed as a compact representation of speech, and can be trained either in a supervised or an unsupervised manner. For the unsupervised case, given an utterance $U = (u_1, u_2, ..., u_n)$, where n is the number of frames, the Gaussian Posteriorgram (GP) for the ith frame is defined as

$$gp_{u_i} = [P(C_1|u_i), P(C_2|u_i), ..., P(C_D|u_i)], \qquad (1)$$

where C_j is a component from a D-component Gaussian mixture model (GMM) which can be trained from a set of unlabeled speech. Zhang et. al [4] explored the use of GPs on unsupervised keyword detection by sorting the alignment scores. Their subsequent work [13] showed that posteriorgrams decoded from Deep Boltzmann Machines can further improve the system performance. Besides the alignment scores, Muscariello et. al [14] also investigated some image processing techniques to compare the self-similarity matrices (SSMs) of two words. By combining the DTW-based scores with the SSM-based scores, the performance on spoken term detection can be improved.

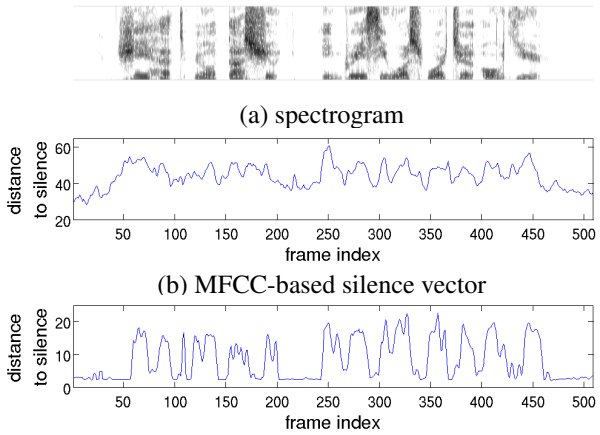
3. WORD SEGMENTATION

Fig. 1 shows the flowchart of our system. Our system detects mispronunciation at the word level, so the first stage is to locate word boundaries in the student's utterance. A common property of nonnative speech is that there can sometimes be a long pause between words. Here we propose incorporating a silence model when running DTW. In this way, we can align the two utterances while also detecting and removing silence in the student's utterance.

Given a teacher frame sequence $T = (f_{t_1}, f_{t_2}, ..., f_{t_n})$ and student frame sequence $S = (f_{s_1}, f_{s_2}, ..., f_{s_m})$, an $n \times m$ distance matrix, Φ_{ts}, can be built, where

$$\Phi_{ts}(i, j) = D(f_{t_i}, f_{s_j}), \qquad (2)$$

and $D(f_{t_i}, f_{s_j})$ denotes any possible distance metric between the speech representation f_{t_i} and f_{s_j}. Here n is the total number of frames of the teacher's utterance and m the student's. If we use Mel-frequency cepstral coefficients (MFCCs) to represent f_{t_i}'s and f_{s_j}'s, $D(f_{t_i}, f_{s_j})$ can be the Euclidean distance between them. If we choose a Gaussian posteriorgram (GP) as the representation, $D(f_{t_i}, f_{s_j})$ can be defined



(a) spectrogram

(b) MFCC-based silence vector

(c) GP-based silence vector

Fig. 2: An example of a spectrogram and the corresponding silence vectors

as $-\log(f_{t_i} \cdot f_{s_j})$ [3, 4]. Given a distance matrix, DTW searches for the path starting from $(1,1)$ and ending at (n,m), along which the accumulated distance is the minimum.

We further define a $1 \times m$ silence vector, ϕ_{sil}, which records the average distance between each frame in S and r silence frames in the beginning of T. In other words, $\phi_{sil}(j)$ records how close f_{s_j} is to silence. ϕ_{sil} can be computed as

$$\phi_{sil}(j) = \frac{1}{r}\sum_{k=1}^{r} D(f_{t_k}, f_{s_j}) = \frac{1}{r}\sum_{k=1}^{r} \Phi_{ts}(k,j). \quad (3)$$

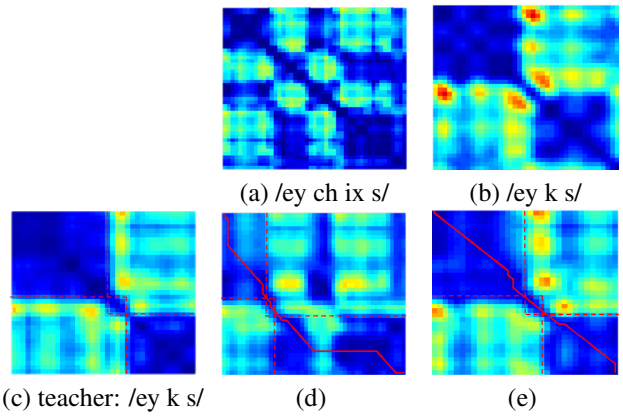
Fig. 2 shows two examples of silence vectors. From the spectrogram we can see that there are three long pauses in the utterance, one starting from the beginning to frame 44, one from frame 461 to the end, and one intra-word pause from frame 216 to frame 245. In the silence vectors, these regions do have relatively low average distance to the first 3 silence frames from a reference utterance.

To incorporate ϕ_{sil}, we consider a modified $n \times m$ distance matrix, Φ'_{ts}. Let B_t be a set of indices of word boundaries in T. Then, each element in Φ'_{ts} can be computed as

$$\Phi'_{ts}(i,j) = \begin{cases} \min(\Phi_{ts}(i,j), \phi_{sil}(j)), & \text{if } i \in B_t \\ \Phi_{ts}(i,j), & \text{otherwise} \end{cases} \quad (4)$$

At word boundaries of the native utterance, Φ'_{ts} would be $\phi_{sil}(j)$ if it is smaller than $\Phi_{ts}(i,j)$, i. e. s_j is closer to silence. DTW can be carried out on Φ'_{ts} to search for the best path. If the path passes through elements in Φ'_{ts} that were from ϕ_{sil}, we could determine that the frames those elements correspond to are pauses.

Locating word boundaries in S is then easy. We first remove those pauses in S according to the information embedded in the aligned path. Then, we map each word boundary in T through the path to locate boundaries in S. If there are multiple frames in S aligned to a boundary frame in T, we choose the midpoint of that segment as the boundary point.



(a) /ey ch ix s/ (b) /ey k s/

(c) teacher: /ey k s/ (d) (e)

Fig. 3: (a) and (b) are the self-similarity matrices of two students saying "aches", (c) is a teacher saying "aches", (d) shows the alignment between (a) and the teacher, and (e) shows the alignment between (b) and the teacher. The dotted lines in (c) are the boundaries detected by the unsupervised phoneme-unit segmentor, and those in (d) and (e) are the segmentation based on the detected boundaries and the aligned path.

4. MISPRONUNCIATION DETECTION

When aligned with a teacher's utterance, a good pronunciation and a bad one will have different characteristics of the aligned path and the distance matrix. Fig. 3-(d) shows the alignment between a teacher and a student who mispronounced the word "aches" as */ey ch ix s/*, while Fig. 3-(e) illustrates the alignment between the same teacher and a student who pronounced the word correctly as */ey k s/*. The most obvious difference between the two should be the high-distance region near the center of Fig. 3-(d), which is the distance between */ch ix/* and */k/*.

In the second stage, we extract features that reflect the degree of mis-alignment. We first propose an unsupervised phoneme segmentor to segment each word into smaller phoneme-like units for a more detailed analysis.

4.1. Unsupervised Phoneme-like Unit Segmentation

Let Φ_{tt} be the self-similarity matrix (SSM) of T, which is generated by aligning T to itself. It should be a square matrix and symmetric along the diagonal (see Fig. 3-(c)). On the diagonal, each low-distance block indicates frames that are phonetically-similar. These frames may relate to a single phoneme, a part of a dipthong, or a chunk of acoustically-similar phonemes. Similar to a music segmentation task [15], we can determine the boundaries in an unsupervised manner by minimizing the sum of the average distance in the lower-triangle of each possible block. Here we denote the unknown

number of segments as K, and the formulation is as follows:

$$(b_0^*, b_1^*, ..., b_{K^*-1}^*, K^*) = \underset{\substack{(b_0,b_1,...,b_{K-1},K) \\ 1=b_0<b_1<...<b_{K-1}\leqslant n \\ 1\leq K\leq n \\ b_K=n+1}}{\operatorname{argmin}} \alpha K + \sum_{z=1}^{K} \sum_{y=b_{z-1}}^{b_z-1} \sum_{x=b_{z-1}}^{y} \frac{\Phi_{tt}(y,x)}{b_z - b_{z-1}}, \quad (5)$$

where $(b_0, b_1, ..., b_{K-1})$ are the starting indices of each segment, n is the length of T, and α is a parameter that is introduced as a penalty term to avoid generating too many segments. The dotted lines in Fig. 3-(c) show a result of segmentation. Together with the aligned path, we can divide each word in Φ_{ts} into several blocks (see the regions bounded by dotted lines in Fig. 3-(d),(e)).

4.2. Feature Extraction

4.2.1. Phone-level features

Several kinds of features have been designed based on the assumption that within each block, if the aligned path is off-diagonal, or the average distance is high, there is a higher probability that the word is mispronounced:

- the ratio of the length of the longest vertical (or horizontal) segment to the length of the aligned path
- the average distance along the aligned path, along the diagonal of the block, and the difference/ratio between the two
- the ratio between the width and the height of the block
- the ratio between the relative width (the width of the block divided by the duration of the word in S) and the relative height (the height of the block divided by the duration of the word in T)
- the average distance within the block
- the difference between the average distance of the block and that of the corresponding block from the SSM of the reference word
- the distance between the average of the speech features within the segment in T and that within the corresponding segment in S

For all of the above features, larger values indicate worse alignment. We pick the maximum value among all segments for each category to form the final phone-level features.

4.2.2. Word-level features

Fig. 3-(a)-(c) are the SSMs from three speakers saying the same word "aches". We can see that a mispronounced version (Fig. 3-(a), with one substitution and one insertion errors) results in a different appearance of the SSM. Muscariello et. al [14] have proposed comparing the structure of two SSMs for the task of keyword spotting, and the structure information was extracted by computing the local histograms of oriented gradients [16]. Similarly, we can adopt this technique to compare two SSMs of the same word. Two speech sequences are first warped into the same length according to the aligned path, and we focus on the SSMs of the warped sequences. Features are extracted as below:

- the average distance along the aligned path, along the diagonal of the distance matrix of the word, and the difference/ratio between the two
- the absolute element-wise difference between the SSMs of the teacher and the student, averaged by the total area
- the absolute difference between the local histograms of oriented gradients of the two SSMs, averaged by the total area
- the average absolute element-wise difference between the two SSMs, only focusing on the blocks along the diagonal, which result from the phoneme-like unit segmentor
- the average absolute difference between the local histograms of oriented gradients of the two SSMs, focusing on the blocks along the diagonal only

The above features, together with the average of the native speech sequence across the word, form the final word-level features.

4.3. Classification

Given the extracted features and a set of good or mispronounced labels, detecting mispronunciation can be treated as a classification task. We adopt libsvm [17] to implement suppport vector machine (SVM) classfiers with an RBF kernel. If there are multiple matching reference utterances, we average the posterior probability output from all the reference speakers to make the final decision.

5. EXPERIMENTS

5.1. Dataset

The nonnative speech comes from the Chinese University Chinese Learners of English (CU-CHLOE) corpus [10], which is a specially-designed corpus of Cantonese speaking English. We use the part of the corpus that is based on TIMIT prompts and divide the 50 male and 50 female speakers into 25 male and 25 female for training, and the rest for testing. Annotations on word-level pronunciation correctness were collected through Amazon Mechanical Turk (AMT) [18]. There were three turkers labeling each utterance, and only words whose labels received agreement among all three turkers (about 87.7% of the data) were used.

Native speech comes from the TIMIT corpus. Only reference speakers of the same gender as the student are used for alignment. We choose the prompts in the *SI* set for training, and *SX* for testing. In the end, the training set consists of 1,196 nonnative utterances, including 1,523 mispronounced words and 8,466 good ones, and the test set consists of 1,065 utterances, including 1,406 mispronounced words and 5,488 good ones. There is only one matching reference utterance

	deviation (frames)	accuracy ($\leq$ 10ms)	accuracy ($\leq$ 20ms)
MFCC	10.1	35.2%	45.2%
GP	9.5	38.2%	47.7%
GP+sil	6.5	41.5%	51.9%
MFCC+sil	6.0	42.2%	53.3%

Table 1: Performance of word segmentation under different scenarios (MFCC: MFCC-based DTW, GP: GP-based DTW, sil: silence model)

for each student's utterance in the training set, compared to 3.8 reference utterances on average in the test set.

All audios are first transformed into 39-dim MFCCs, including first and second order derivatives, at every 10-ms frame. A 150-mixture GMM is trained on all TIMIT utterances for GP decoding.

5.2. Word Segmentation

We first examine how well DTW can capture word boundaries. The nonnative data in both the training set and the test set are used for evaluation. Ground truth timing information on the nonnative data is generated through forced alignment. The size of the silence window, r, is chosen to be 3 for computing ϕ_{sil}. We compute the absolute deviation between the ground truth and the detected boundary, and the percentage of boundaries that fall within a 10-ms or 20-ms window from the ground truth. If there is more than one reference native utterance for an utterance, the one that gives the best performance is considered.

Four scenarios are tested as shown in Table 1. With the help of the silence model, MFCC-based DTW obtains a 40.6% relative improvement and GP-based DTW has a 31.6% relative improvement in terms of deviation in frames. In both cases, more than half of the detected word boundaries are within a 20-ms window to the ground truth.

The silence model helps both GP and MFCC-based approaches due to the significant amount of silence between words in the nonnative data, which takes up 37.0% of the total time duration. The MFCC-based silence model can capture 77.4% of the silence with a precision of 90.0%, and the GP-based silence model can capture 72.3% of the silence with a precision of 86.1%. Both models can detect most of the silence frames with high precision, and thus, the word boundaries can be more accurately captured. One possible explanation for the slightly lower performance of GP-based DTW is that there is more than one mixture in the unsupervised GMM that captures the characteristics of silence, so silence frames in different utterances may have different distributions over the mixtures.

5.3. Mispronunciation Detection

Here we examine the performance of the proposed framework on mispronunciation detection. *Precision*, *recall* and *f-score* are used for evaluation. *Precision* is the ratio of the number of words that are correctly identified by the classifier as mispronounced to the total number of hypothesized mispronunciations, *recall* is the ratio of the number of mispronounced words that are correctly detected to the total number of mispronounced words in the data, and *f-score* is the harmonic mean of the two. The parameters of the SVM are optimized for different scenarios, respectively.

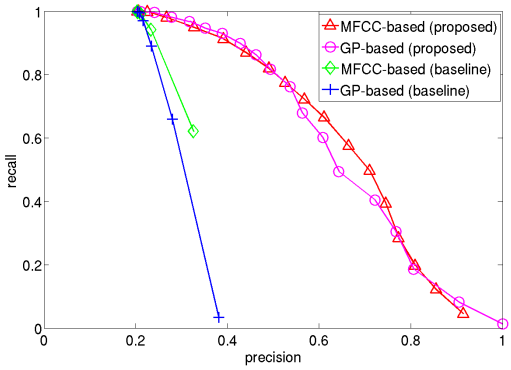


Fig. 4: ROC curves of differenct scenarios

f-score (%)	MFCC-based	GP-based
overall	63.7	63.0
baseline	42.8	39.3
phone-level	59.1	61.4
word-level	61.3	60.0

Table 2: Overall system performance and the performance of using different levels of features

5.3.1. *System performance*

For the baseline, we build a naive framework with only a subset of word-level features, which are the average distance along the aligned path, the average distance along the diagonal of the distance matrix of the word, and the difference/ratio between the two. In other words, the baseline considers the word-level alignment scores only.

Fig. 4 shows the ROC curves of the overall system performance and the baseline performance based on either MFCC-based or GP-based alignment, and the first two rows in Table 2 summarize the results of the best f-score in each scenario. Our proposed framework improves the baseline by at least 49% in f-score relatively. This shows that merely considering the distance along the aligned path is not enough. Extracting features based on the shape of the aligned path or the appearance of the distance matrix, or segmenting a word into subword units for more detailed analysis, can give us more information about the quality of the alignment, and thus the quality of the pronunciation.

The MFCC-based framework performs slightly better than the GP-based one. However, the difference is not statistically significant ($p > 0.1$ using McNemar's test). There are many factors affecting the overall performance. For ex-

ample, after randomly sampling some annotations collected from AMT, we found a subset of them to be problematic, even though all three turkers had agreement. This lowers the quality of the training data.

5.3.2. Different levels of features

The last two rows in Table 2 show the system performance based on either word-level or phone-level features only. Compared with the baseline, a system with word-level features only can achieve a relative increase of around 45%. This again shows the benefits of having features that compare the structure of the distance matrices. A system with phone-level features only can also improve the performance by 47% relative to the baseline. We can see that combining the features from different levels did help improve performance. The improvement is statistically significant with $p < 0.001$ using McNemar's test, which indicates that the features from the two levels have complementary information to one another. By further combining word-level, MFCC-based features with phone-level, GP-based features, the overall performance can be improved to an f-score of 65.1% ($p < 0.001$ compared with an MFCC-based system). This result implies that not only do word-level and the phone-level features have complementary information, but MFCC-based and GP-based features can also be combined to boost performance.

6. CONCLUSION AND FUTURE WORK

In this paper, we present our efforts to build a mispronunciation detection system that analyzes the degree of misalignment between a student's speech and a teacher's without requiring linguistic knowledge. We show that DTW works well in aligning native speech with nonnative speech and locating word boundaries. Such results suggest that many keyword spotting approaches may be able to work on nonnative speakers. Features that capture the characteristics of an aligned path and a distance matrix are introduced, and the experimental results show that the system outperforms the one that considers alignment scores only.

Though it is commonly acknowledged that phone-level feedback has higher pedagogical values than word-level feedback, we believe that for low-resource languages, providing word-level feedback is a proper first step towards detecting pronunciation errors at finer granularity. Several issues remain to be explored. First, some parts of the mis-alignment come from the differences in the non-linguistic conditions of the speakers, e. g. vocal tracts or channels. One next step would be to consider phonology features that are more robust to different speaker characteristics. Also, it would be interesting to explore system performance on other target languages, or with students from different native languages.

7. REFERENCES

[1] "Ethnologue: Languages of the world," http://www.ethnologue.com/.

[2] "Nuance recognizer language availability," http://www.nuance.com/for-business/by-solution/customer-service-solutions/solutions-services/inbound-solutions/self-service-automation/recognizer/recognizer-languages/index.htm.

[3] T.J. Hazen, W. Shen, and C. White, "Query-by-example spoken term detection using phonetic posteriorgram templates," in *ASRU*, 2009, pp. 421–426.

[4] Y. Zhang and J.R. Glass, "Unsupervised spoken keyword spotting via segmental DTW on gaussian posteriorgrams," in *ASRU*, 2009, pp. 398–403.

[5] D. Kewley-Port, C. Watson, D. Maki, and D. Reed, "Speaker-dependent speech recognition as the basis for a speech training aid," in *ICASSP*, 1987, vol. 12, pp. 372–375.

[6] SM Witt and SJ Young, "Phone-level pronunciation scoring and assessment for interactive language learning," *Speech communication*, vol. 30, no. 2, pp. 95–108, 2000.

[7] H. Franco, L. Neumeyer, M. Ramos, and H. Bratt, "Automatic detection of phone-level mispronunciation for language learning," in *Sixth European Conference on Speech Communication and Technology*, 1999.

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

[9] N. Minematsu, S. Asakawa, and K. Hirose, "Structural representation of the pronunciation and its use for call," in *SLT*, 2006, pp. 126–129.

[10] H. Meng, Y.Y. Lo, L. Wang, and W.Y. Lau, "Deriving salient learners' mispronunciations from cross-language phonological comparisons," in *ASRU*, 2007, pp. 437–442.

[11] A.M. Harrison, W.Y. Lau, H.M. Meng, and L. Wang, "Improving mispronunciation detection and diagnosis of learners' speech with context-sensitive phonological rules based on language transfer," in *Ninth Annual Conference of the ISCA*, 2008.

[12] Y.-B. Wang and L.-S. Lee, "Improved approaches of modeling and detecting error patterns with empirical analysis for computer-aided pronunciation training," in *ICASSP*, 2012.

[13] Y. Zhang, R. Salakhutdinov, H. Chang, and J. Glass, "Resource configurable spoken query detection using deep Boltzmann machines," in *Proc. ICASSP*, 2012.

[14] A. Muscariello, G. Gravier, and F. Bimbot, "Towards robust word discovery by self-similarity matrix comparison," in *Proc. ICASSP*, 2011.

[15] Kristoffer Jensen, "Multiple scale music segmentation using rhythm, timbre, and harmony," *EURASIP Journal on Advances in Signal Processing*, vol. 2007, pp. 1 – 11, 2007.

[16] N. Dalal and B. Triggs, "Histograms of oriented gradients for human detection," in *CVPR*, 2005, vol. 1, pp. 886–893.

[17] Chih-Chung Chang and Chih-Jen Lin, "LIBSVM: A library for support vector machines," *ACM Transactions on Intelligent Systems and Technology*, vol. 2, pp. 27:1–27:27, 2011, Software available at http://www.csie.ntu.edu.tw/∼cjlin/libsvm.

[18] Mitchell A. Peabody, *Methods for pronunciation assessment in computer aided language learning*, Ph.D. thesis, MIT, 2011.