Available online at www.sciencedirect.com

ScienceDirect

Speech Communication 67 (2015) 154–166

SPEECH COMMUNICATION

www.elsevier.com/locate/specom

Improved mispronunciation detection with deep neural network trained acoustic models and transfer learning based logistic regression classifiers

Wenping Hu [a,b], Yao Qian [b,*], Frank K. Soong [b], Yong Wang [a]

[a] University of Science and Technology of China, Hefei 230026, China
[b] Microsoft Research Asia, Beijing 100080, China

Received 28 July 2014; received in revised form 19 December 2014; accepted 27 December 2014
Available online 8 January 2015

Abstract

Mispronunciation detection is an important part in a Computer-Aided Language Learning (CALL) system. By automatically pointing out where mispronunciations occur in an utterance, a language learner can receive informative and to-the-point feedbacks. In this paper, we improve mispronunciation detection performance with a Deep Neural Network (DNN) trained acoustic model and transfer learning based Logistic Regression (LR) classifiers. The acoustic model trained by the conventional GMM-HMM based approach is refined by the DNN training with enhanced discrimination. The corresponding Goodness Of Pronunciation (GOP) scores are revised to evaluate pronunciation quality of non-native language learners robustly. A Neural Network (NN) based, Logistic Regression (LR) classifier, where a general neural network with shared hidden layers for extracting useful speech features is pre-trained firstly with pooled, training data in the sense of transfer learning, and then phone-dependent, 2-class logistic regression classifiers are trained as phone specific output layer nodes, is proposed to mispronunciation detection. The new LR classifier streamlines training multiple individual classifiers separately by learning the common feature representation via the shared hidden layer. Experimental results on an isolated English word corpus recorded by non-native (L2) English learners show that the proposed GOP measure can improve the performance of GOP based mispronunciation detection approach, i.e., 7.4% of the precision and recall rate are both improved, compared with the conventional GOP estimated from GMM-HMM. The NN-based LR classifier improves the equal precision–recall rate by 25% over the best GOP based approach. It also outperforms the state-of-art Support Vector Machine (SVM) based classifier by 2.2% of equal precision–recall rate improvement. Our approaches also achieve similar results on a continuous read, L2 Mandarin language learning corpus.

Keywords: Computer-aided language learning; Mispronunciation detection; Deep neural network; Logistic regression; Transfer learning

1. Introduction

The current globalization of people in regions of different languages has accelerated the demand of foreign language proficiency. For non-native language learners, the one-to-one teacher–student interaction and communication is the most effective way, but it can be too pricey and unaffordable for many learners. Computer Aided Language Learning (CALL) systems, powered by the advancement of speech technology, can bridge the gap between disproportional supply and demand in language learners and teachers and have become ubiquitous learning tools with handy smart phones, tablets, laptop computers, etc. However, as an indispensable component of CALL system, phone level mispronunciation detection, which aims at

* Corresponding author. Tel.: +86 10 59174292.
E-mail addresses: hwping@mail.ustc.edu.cn (W. Hu), yaoqian@microsoft.com (Y. Qian), frankkps@microsoft.com (F.K. Soong), yongwang@ustc.edu.cn (Y. Wang).

http://dx.doi.org/10.1016/j.specom.2014.12.008
0167-6393/

detecting or identifying pronunciation errors or deficiency at the phone level in a high precision, is still challenging.

There are a great deal of research work on mispronunciation detection. An in-depth review of automatic error detection in pronunciation training is given by Witt (2012), in which pronunciation errors are divided into two types, i.e., phonemic error and prosodic error, and the phonemic error is further categorized into many sub error types. Features used for detecting the pronunciation errors are mostly extracted from the output of an HMM based speech recognizer. Kim et al. (1997) compare three HMM based scores, e.g., log-likelihood score, log-posterior probability score and segment duration score, in pronunciation quality evaluation for some specific phones and find log-posterior probability scores have the highest correlation with human ratings. Besides this HMM based log-posterior probability based method, Franco et al. (1999) further adopt the Log-Likelihood Ratio (LLR) between native-like and non-native models as the measure for mispronunciation detection. The results show that LLR based method has better overall performance than the posterior based method, but it needs to be trained with specific examples of the target non-native user population. Witt and Young (2000) introduce a Goodness Of Pronunciation (GOP) method, a variation of the posterior probability, for phone level pronunciation scoring. This GOP measure is later widely used in pronunciation evaluation and mispronunciation detection tasks. Some variations of GOP measure are also proposed in the last decade. Zhang et al. (2008) propose a Scaling Log-Posterior Probability (SLPP) method for Mandarin mispronunciation detection and achieve considerable performance improvement. Wang and Lee (2012) also combine the GOP based method with error pattern detectors for phone mispronunciation diagnosis with a serial and parallel structure and find that the serial structure can reduce the average error rate and improve diagnosis feedback. To improve the scores generated by the traditional GMM-HMM based speech recognizer, some discriminative training algorithms have been applied, e.g. Maximum Mutual Information Estimation (MMIE) (Bahl et al., 1986), Minimum Classification Error (MCE) (Juang et al., 1997) and Minimum Phone Error (MPE) and Minimum Word Error (MWE) (Povey and Woodland, 2002). Yan and Gong (2011) introduce the discriminatively refined acoustic models by MPE into pronunciation proficiency evaluation. Qian et al. (2010) also investigate using MWE-trained HMM models to minimize mispronunciation detection errors for L2 English learners.

Mispronunciation detection can be formulated as a 2-class classification task. Many classifier based approaches are applied to mispronunciation detection to improve its performance. Ito et al. (2005) use a decision tree based method to set thresholds for different kinds of mispronunciations and achieve a significant improvement, compared with a universal threshold. Truong (2004) uses decision tree and Linear Discriminant Analysis (LDA) to distinguish different pronunciation errors of L2 learners of Dutch based on some discriminative features, e.g., formants and durations. A specific classifier is built for each mispronunciation pattern. Experimental results show that the classifier based approach has a good performance of detecting vowel pronunciation errors but a poor performance of detecting consonant pronunciation errors. It also shows that LDA yields a better detection performance than decision tree. Strik et al. (2009) draw a comparison of four different approaches, i.e., GOP score, decision tree and LDA with two kinds of features, acoustic–phonetic features and Mel-Frequency Cepstrum Coefficients (MFCCs), in distinguishing two Dutch phones: the velar fricative/x/ and the velar plosive/k/. The result shows that LDA based methods outperform the GOP and decision tree based methods. Doremalen et al. (2009) build a set of Support Vector Machine (SVM) classifiers to detect substitution errors for Dutch vowels and improve the performance by combining different features: confidence measures (Jiang, 2005), phonetic features and MFCCs. Wei et al. (2009) also apply SVM to classify the correct and incorrect pronunciations of Mandarin syllables and improve the performance by enhancing the discrimination of pronunciation variation with Pronunciation Space Models (PSMs). Other research work, which applies SVM as the underlying classifier for mispronunciation detection in different scenarios, can be found in (Jie and Xu, 2009; Xu et al., 2009; Yoon et al., 2010; Hirabayashi and Nakagawa, 2010). Additionally, to improve the accuracy of pronunciation error detection, the knowledge of L1 (the native language), e.g. the set of common pronunciation errors made by non-native speakers, can be used to contrast L1–L2 phone confusion pairs to build a more custom-made mispronunciation detection system.

Recently, Deep Neural Network (DNN), which attempts to model high-level abstractions in data, has significantly improve the discrimination of acoustic models in speech recognition (Hinton et al., 2012a). Transfer learning (Bengio et al., 2013), which can exploit commonalities between different learning tasks in order to share statistical strength, is successfully employed to multilingual speech recognition (Huang et al., 2013). Application of using Deep Belief Nets (DBN) to mispronunciation detection and diagnosis in L2 English has been tried by Qian et al. (2012), and a significant improvement on word pronunciation relative error rate was obtained on L1 (Cantonese)-dependent English learning corpus. We have used DNN trained acoustic models for English pronunciation quality scoring. We find the GOP scores estimated from DNN outputs correlate better with human expert's evaluations than conventional GOP scores obtained from a conventional GMM based system (Hu et al., 2013). We have also investigated different pitch embedding methods in DNN acoustic model training to further enhance the discrimination of acoustic models and detect the pronunciation errors caused by misusing lexical stress or lexical tone for L2 language learners (Hu et al., 2014a). In this study, we focus on the detection of phonemic error, i.e., phone-level mispronunciation. We

propose to use DNN and transfer learning to a task of detecting pronunciation errors on the phone level (Hu et al., 2014b). Multi-layer neural networks are trained as nonlinear basis functions to represent speech signals while the top layer of the network is trained discriminatively to sharpen the posterior probabilities of senones (tied subphonemic states) in DNN based acoustic model training. When building phone mispronunciation classifiers, different from the conventional, separately trained phone specific classifiers, multiple phone specific, 2-class classifiers are built and jointly trained on top of a general neural network with multiple hidden layers. This shared hidden layer structure also enables to learn a better feature representation for mispronunciation diagnosis. In addition, we focus on L1 independent mispronunciation detection system, which is usually preferable for commercial implementation convenience, e.g., online web service. The rest of this paper is organized as follows: Section 2 introduces our proposed approach to mispronunciation detection. The construction of different mispronunciation detection systems is described in Section 3. Section 4 presents the experiments and results. In Section 5, we further evaluate our proposed approaches on Mandarin mispronunciation detection. The implementation of a realtime system is introduced in Section 6. Conclusions are drawn in Section 7.

2. Automatic detection of phone-level mispronunciation

In the last few decades, conventional approaches to mispronunciation detection are mainly GOP based or classifier based. The schematic diagram of these two approaches is shown in Fig. 1.

An L2 learner's utterance and its canonical transcription are firstly fed into acoustic models to get the probabilities or likelihoods, which indicate how similar the learner's pronunciation to the canonical pronunciation is, and other by-product information, e.g., phone duration (obtained by forced alignment). For an L1-independent English CALL system, where the L1 (the native language) knowledge of learners is unknown, native canonical English speakers' corpora are generally used for building acoustic models. Then these outputs of acoustic models can be used to calculate GOP scores for detecting mispronunciation by a pre-set threshold directly (GOP based approach) or used as input features to train 2-class (correct or not) classifier for identifying mispronunciation (classifier based approach). We improve the performance of automatic detection of phone-level mispronunciation by enhancing three key modules: acoustic models, GOP calculator and classifier. Firstly, we enhance the discrimination of acoustic model by DNN training; then we revise the GOP calculation algorithm to robustly evaluate pronunciation quality of non-native learners; finally, a Neural Network (NN) based, Logistic Regression (LR) classifier is proposed to improve the classification performance and generalization capability.

2.1. DNN based acoustic model training

A deep neural network is a feed-forward, artificial neural network with multiple hidden layers between its input and output. For each hidden unit j, a function, typically a logistic one, is used to map all inputs from the lower layer, x_j, to a scalar state, y_j, which is then fed to the upper layer.

$$y_j = logistic(x_j) = \frac{1}{1+e^{-x_j}}; \quad x_j = b_j + \sum_i y_i w_{ij} \tag{1}$$

where b_j is the bias of unit j; i, the unit index of lower layer; w_{ij}, the weight on the connection between unit j and unit i in the layer below. For multi-class classification, a softmax (a generalization of logistic function for probabilistic multi-class classification) nonlinear function is used to convert the inputs, x_j, into a class probability, p_j, given in Eq. (2), where k is an index over all classes.

$$p_j = \frac{\exp(x_j)}{\sum_k \exp(x_k)} \tag{2}$$

All weights and biases are initialized in DBN pre-training (Hinton et al., 2006), and then discriminatively trained by optimizing the cross entropy between the target probability and actual output of softmax with Back-Propagation (BP) procedure (Rumelhart et al., 1986). When softmax output is used, the natural cost function is the cross entropy between the target probability, d_t, and actual output of softmax, p_t. When binary, one or zero, target probability, d_t, is taken, the cost function C can be simplified as:

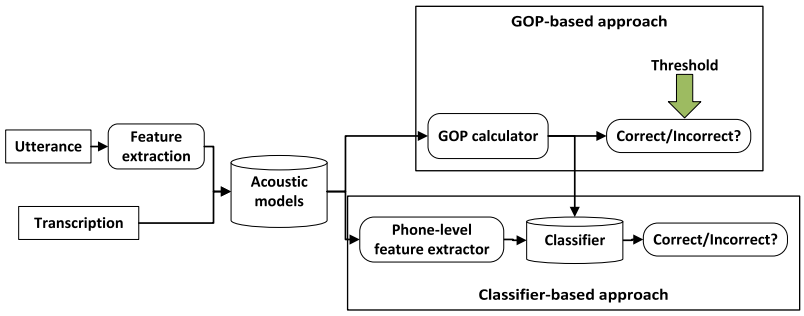


Fig. 1. Framework for mispronunciation detection systems.

$$C = -\sum_{t=1}^{T} d_t \log(p_t) = -\sum_{t=1}^{T} \log P(s_t|\mathbf{o}_t) \quad (3)$$

where $\mathbf{o}_t$ is the training sample; s_t, the corresponding ground truth label; and T is the size of training samples. It is optimized by a "mini-batch" based stochastic gradient descent algorithm. More training details, considerations and the integration of DNN with Context Dependent HMMs can be found in (Hinton et al., 2006; Seide et al., 2011; Dahl et al., 2012).

Comparing with the conventional GMM-HMM based approach, DNN based approach can: model long-span (e.g., 11 frames), high dimensional and strongly correlated features as its input feature; find a highly non-linear mapping between input and output features with a deep-layered hierarchical structure; train model parameters discriminatively through back-propagation with the gradient of a cost function. Further, there is no underlying assumption of distribution and modality for input data in the DNN, e.g., continuous and binary features can be augmented and modeled together naturally by slightly extending Hinton's standard DBN pre-training method (Hinton et al., 2006).

Lexical stress in some languages, e.g., English, and lexical tone in tonal languages, e.g., Chinese, are largely unpredictable, needing to be "learned" for each individual word. They are very difficult but must be mastered for language learners. To detect such pronunciation errors from the utterances of language learners, fundamental frequency (F0) contour, the main acoustic cue of lexical stress and tone, need to be included in acoustic modeling. As we know, different from spectral features, F0 is quasi-continuous and disappears in unvoiced segments. Some heuristic approaches, e.g., interpolating F0 in unvoiced regions, or multi-space distribution (MSD) (Tokuda et al., 2002) are employed to model F0 under GMM-HMM framework. Benefiting from DNN structure, F0 features can be directly appended to the spectral features for modeling speech under DNN-HMM framework (Hu et al., 2014a). In addition, DNN can model stress and tone features, which are suprasegmental features going far beyond the time span of a single frame, more efficiently and effectively since a longer window of observations can be used to capture the supersegmental characteristics under DNN-HMM framework.

2.2. Goodness Of Pronunciation (GOP) calculation

In the conventional GMM-HMM based system, the GOP score of phone p given the whole observations $\mathbf{o}$, first proposed by Witt and Young (2000), is defined as:

$$\begin{aligned} \mathrm{GOP}(p) &= \frac{1}{t_e - t_s + 1} * \log p(p|\mathbf{o}) \\ &= \frac{1}{t_e - t_s + 1} * \log \frac{p(\mathbf{o}|p)p(p)}{\sum_{\{q \in Q\}} (p(\mathbf{o}|q)p(q)} \end{aligned} \quad (4)$$

where t_s and t_e are the start and end frame indexes, respectively; Q is the whole phone set; $p(p)$ is the prior of phone p. The numerator of Eq. (4) is calculated from forced alignment result and the denominator is calculated from an output lattice, generated from automatic speech recognition with an unconstrained phone loop (Witt and Young, 2000). In practice, we use the Generalized Posterior Probability (GPP) (Soong et al., 2004) method, which relaxes unit boundary to avoid underestimating of posterior probability in a reduced search space, i.e. a lattice, in above GOP score calculation.

In our DNN-HMM based system, the log phone posterior ratio between the canonical phone and the one with the highest score is used to approximate GOP score. For phone p, it is defined as:

$$\mathrm{GOP}(p) \approx \log \frac{p(p|\mathbf{o}; t_s, t_e)}{\max_{\{q \in Q\}} p(q|\mathbf{o}; t_s, t_e)} \quad (5)$$

where Q is whole phone set.

As the output, softmax, layer of DNN based acoustic model can directly output the posterior of each senone s_t, a straight way to approximate the Log Phone Posterior (LPP) of phone segment p, i.e., $\log p(p|\mathbf{o}; t_s, t_e)$, given the whole segment $\mathbf{o}$, is by averaging the frame based posterior $p(s_t|\mathbf{o})$. The posterior of phone p is defined as

$$\mathrm{LPP}(p) = \log p(p|\mathbf{o}; t_s, t_e) \approx \frac{1}{t_e - t_s + 1} \sum_{t=t_s}^{t_e} \log p(s_t|\mathbf{o}_t) \quad (6)$$

where $\mathbf{o}_t$ is the augmented input observations of the frame t; s_t is the senone label of the frame t generated by force alignment with the given canonical phone p; $p(s_t|\mathbf{o}_t)$ is the output of the last softmax layer of DNN acoustic model. It can avoid the decoding lattice and its corresponding forward–backward computations, needed for GOP calculation in Eq. (4).

The frame based posterior score, $p(s_t|\mathbf{o}_t)$ in Eq. (6), is calculated as the softmax output of a given senone, obtained by forced-aligning the utterance of language learners with the corresponding canonical phone sequence. Our acoustic model is trained by native speakers' utterances, which are uttered in a standard English way, while the utterances from language learners generally have some accents. Therefore, there is a mismatch between the training native speakers' utterances and testing non-native speakers' utterances and this mismatch will result in inaccurate forced-alignment results at senone level. To robustly evaluate pronunciation quality of non-native learners, we revise the GOP calculation as:

$$\mathrm{LPP}(p) = \log p(p|\mathbf{o}; t_s, t_e) \approx \frac{1}{t_e - t_s + 1} \sum_{t=t_s}^{t_e} \log p(p|\mathbf{o}_t) \quad (7)$$

$$p(p|\mathbf{o}_t) = \sum_{s \in p} p(s|\mathbf{o}_t) \quad (8)$$

where s is the senone label, $\{s|s \in p\}$ is the set of all senones corresponding to p, i.e., the states belonging to those

triphones (HMM models) whose current phone is p. In Eq. (8), all possible senones are considered in GOP score calculation, i.e., the senone boundaries are relaxed to phone boundaries.

2.3. Neural network based logistic regression classifiers

We build multiple phone specific 2-class logistic regression classifiers above a shared, common neural network for identifying mispronunciation errors on phone level.

2.3.1. 2-class logistic regression

Logistic regression is a probabilistic statistical classification method, which has been widely applied to two class classification tasks and it is described in detail in (Bishop, 2006). The phone mispronunciation detection problem can be formulated as a classical, 2-class classification task, i.e., Correct (C) or Incorrect (IC). As a discriminative model, logistic regression classifier learns the conditional probability directly. Given the sample vector x, the posterior probability of class C is modeled as a logistic sigmoid function:

$$p(C|x) = y(x) = \sigma(\mathbf{w}^T x); \quad \sigma(z) = \frac{1}{1+e^{-z}} \tag{9}$$

with $p(IC|x) = 1 - p(C|x)$. Here $\sigma(\cdot)$ is the logistic sigmoid function, $\mathbf{w}$ is the set of weights to be learned.

For data set $\{x_n, t_n\}_{n=1}^N$, where x_n is the n-th sample vector, $t_n \in \{0,1\}$ is the corresponding reference label, 1 for correct and 0 for incorrect. The likelihood function of the outcome can be written as

$$L = \prod_{n=1}^{N} y_n^{t_n}\{1-y_n\}^{1-t_n}; \quad y_n = \sigma(\mathbf{w}^T x_n) = p(C|x_n) \tag{10}$$

By taking the negative logarithm of this likelihood function, we obtain the cross-entropy error function:

$$E(\mathbf{w}) = -\ln L = -\sum_{n=1}^{N}\{t_n \ln y_n + (1-t_n)\ln(1-y_n)\} \tag{11}$$

Gradient descent algorithm can be used to optimize the error function, the gradient of the error function with respect to $\mathbf{w}$ is:

$$\nabla E(\mathbf{w}) = \sum_{n=1}^{N}(y_n - t_n)x_n \tag{12}$$

2.3.2. Multiple 2-class logistic regression classifiers

In our phone-level mispronunciation detection task, there are M different phones and each sample vector belongs to only one of them. We group the whole data set into M subsets by its phone label. For each subset, a 2-class logistic regression classifier is trained for the correct or incorrect classification.

Instead of training each phone specific classifier separately, we learn all these M classifiers jointly in a neural network based framework. Different from the conventional neural network, the last output layer consists of M logistic regression classifiers, each for classifying one phone. The framework of our neural network based logistic regression classifiers is shown in Fig. 2. The likelihood function is written as:

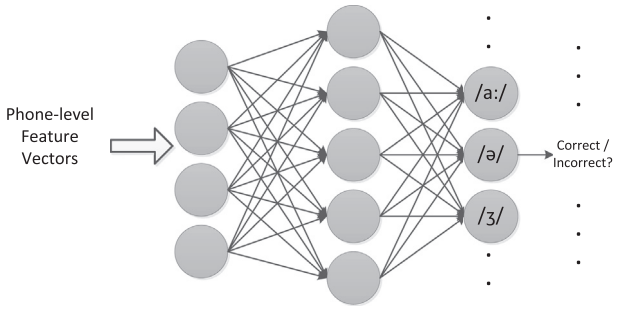


Fig. 2. Overview of framework: neural network based logistic regression classifier.

$$L = \prod_{n=1}^{N} y_n^{t_n}\{1-y_n\}^{1-t_n}; \quad y_n = \sigma_i(\mathbf{w}_i^T \phi_n) \quad \phi_n = \phi(x_n); \tag{13}$$

where $i \in \{1,\ldots,M\}$ is the phone index that sample x_n belongs to; $\mathbf{w}_i$ and σ_i are the corresponding weights and logistic sigmoid function, respectively, of that specific classifier; $\phi(\cdot)$ is the nonlinear function, provided by this shared neural network; ϕ_n is the transformed feature vector of x_n. In general, multiple phone specific 2-class logistic regression classifiers are built above a general neural network with multiple hidden layers.

The training recipe is briefly described as follows: Firstly, all the weights and biases of hidden layers are initialized with stacked Restricted Boltzman Machines (RBMs) (Hinton et al., 2006) with all pooled, training data from different phones. This stage is known as pre-training. Secondly, multiple phone specific 2-class logistic regression classifiers are added above the hidden layers with randomly initialized weights. Finally, all the parameters, including LR classifiers and hidden layers, are discriminatively trained by minimizing the negative logarithm likelihood, described in Eq. (13), with Back-Propagation (Rumelhart et al., 1986) in a "mini-batch" based stochastic gradient descent algorithm. From Eq. (13), we can see that, for each sample, only one of these M classifiers is activated and only the likelihood from that activated classifier is calculated in the final fine-tuning phase. It means that each LR classifier is trained only with its phone specific data but the hidden layers are trained with all the pooled, training data both in pre-training and fine-tuning phase. Through this shared hidden layer structure and training approach, we expect to learn a better feature representation for improving the performance of mispronunciation detection. To improve the generalibility, dropout technique (Hinton et al., 2012b) is adopted in this network training.

In testing, for an input phone-level feature vector x, belonging to phone subset i, the decision is

$$p(C|\phi(x)) = \sigma_i(\mathbf{w}_i^T\phi(x)) \begin{cases} \geqslant \theta & C: \text{ Correct} \\ < \theta & IC: \text{ Incorrect} \end{cases}$$

where θ is a threshold. By default it is set as 0.5 for our NN-based LR classifier.

By training all the samples in a joint neural network, it streamlines training multiple individual classifiers separately, i.e., one for each specific phone. In addition, with this shared hidden layer structure, it not only helps to extract more predictive features to distinguish correct or incorrect phones, but also improves the generalibility for some phones which has few samples to build a good independent classifier. These properties are verified in our following experiments. This approach can be seen as a kind of transfer learning (Bengio et al., 2013). Building the classifiers for different phones can be regarded as multi-task application. By sharing hidden layers, it can exploit commonalities between different phone mispronunciation detection tasks to enhance some classifiers which have insufficient training samples.

3. Mispronunciation detection system construction

We build three GOP based systems and two classifier based systems for mispronunciation detection.

3.1. GOP based system

We first train a context dependent GMM-HMM acoustic model by using Expectation Maximization (EM) algorithm with native U.S. English speakers' recordings and their corresponding transcriptions. This model is used to obtain the senone label for each frame by forced-aligning the utterance with its corresponding transcription. Then these aligned senone sequences and frame sequences are used as pairs of input and output features to train a DNN-HMM acoustic model, as described in Section 2.1, in a sense of discriminative training. To build GOP based mispronunciation detection systems, GOP of each phone segment is calculated by equations described in Section 2.2, after forced-aligning the language leaner's utterance (frame sequence) with the canonical transcription which the learner should utter. A unique phone independent threshold (single threshold) or multiple phone dependent thresholds are determined based on the human labeled mispronunciation data.

Three different GOP based systems are built to evaluate our proposed methods. They include one GOP measure defined as Eq. (4) in GMM-HMM based systems, denoted as GOP-GMM, and two GOP measures defined as Eq. (5) in DNN-HMM based systems, denoted as GOP-DNN1 and GOP-DNN2, whose phone segmental posterior is calculated as Eqs. (6) and (7), respectively.

3.2. Classifier based systems

SVM classifier and our proposed NN-based LR classifier are built for mispronunciation detection. Different from GOP based method, an extra supervised training process is needed. The input features for supervised training are phone-level, segmental features. As illustrated in Fig. 3, these features of each segment $\mathbf{o}_i$ are calculated from the frame posterior matrix, outputs of DNN, with its start and end frame indexes (t_s, t_e), obtained by forced-alignment. The Log Phone Posterior (LPP) of any phone p is defined as Eq. (7) and Log Posterior Ratio (LPR) between phone p_j and p_i is defined as:

$$\text{LPR}(p_j|p_i) = \log p(p_j|\mathbf{o}; t_s, t_e) - \log p(p_i|\mathbf{o}; t_s, t_e) \tag{14}$$

The segmental feature vector $f(p_i|\mathbf{o}; t_s, t_e)$ of canonical phone p_i is defined as follows:

$$\begin{aligned} f(p_i|\mathbf{o}; t_s, t_e) = [&\text{LPP}(p_1), \text{LPP}(p_2), \ldots, \text{LPP}(p_j), \ldots, \text{LPP}(p_M), \\ &\text{LPR}(p_1|p_i), \text{LPR}(p_2|p_i), \ldots, \\ &\text{LPR}(p_j|p_i), \ldots, \text{LPR}(p_M|p_i)]^T \end{aligned} \tag{15}$$

where M is the total number of all phones.

Each phone segment is manually labeled as correct or incorrect pronunciation, which is used as the output labels. SVM classifiers or NN-based LR classifiers are trained based on the input and output features of training data. For SVM classifier training, multiple individual 2-class SVM classifiers are trained separately (each phone has a classifier), with the tool SVM-light (Joachims, 1998; Vapnik, 1995), where linear kernel is used here. The training process of our NN-based LR classifier is described in Section 2.3 and we implement it based on Seide's DNN training platform (Seide et al., 2011).

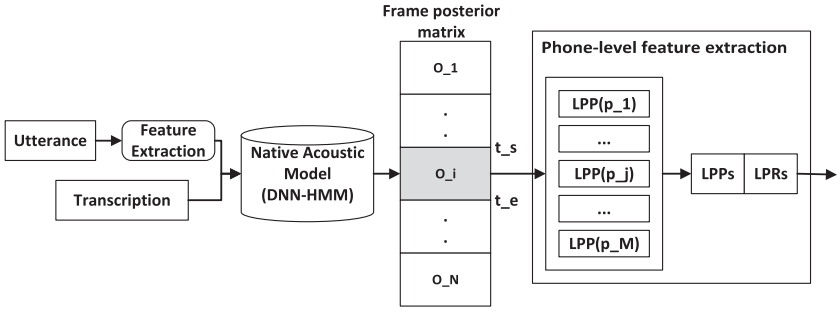


Fig. 3. Flowchart for phone-level feature extraction.

4. Experiments and results

4.1. Databases

Two databases are used in our experiments. One is recorded by native speakers (native database), which is used to train the native acoustic model. And the other one is recorded by L2 language learners (non-native database), used to train and evaluate the performance of different classifiers for mispronunciation detection.

4.1.1. Native database

In this study, a phonetically rich, isolated word, telephone speech corpus, named "LDC95S27" (Pitrelli and Fong, 1995), is used to train the native acoustic model for phone mispronunciation detection. Each utterance contains a single isolated word. The full training set consists of 90 word lists, each list contains 75 distinctive words and each word is spoken by ~10 speakers. Neither the speaker nor word is mixed distributed across different lists. The training set contains 900 speakers, ~6.7k distinct words, ~20 h data in total. Another 8 word lists, which contain 5k words, 80 speakers in total, are used to evaluate the discriminability of acoustic models by a speech recognition task.

4.1.2. Non-native database

To evaluate the performance of mispronunciation detection, a read English, isolated word corpus is recorded by 60 non-native English learners (all Chinese) with different spoken English proficiency levels, based on their TOFEL[1] or IELTS[2] oral scores. Each speaker records ~300 words, whose transcriptions are randomly selected from the "LDC95S27" word corpus. The "ground truth" assessments of pronunciation errors are obtained by one linguistic expert. 43 learners' data, selected from different proficiency bands according to their spoken English proficiency scores, is used for training, 7 learners' data is used for validation set to tune the hyper-parameters of classifiers and the remaining 10 speakers' data is used to evaluate the performance of different detection approaches. The expert marks the phone insertion, deletion and substitution errors for each word of each speaker. The number of substitution, deletion and insertion error tokens in the whole database is shown in Table 1. It shows that among all these three error types, substitution error is dominant. As a preliminary study, we only focus on the substitution errors in this study. The detection of insertion and deletion errors will be investigated in our future work. The number of correct and incorrect (only substitution errors are considered) tokens in training, validation and test sets is shown in Table 2. The mispronunciation rate, the percentage of incorrect phone tokens in all the data set, is about 13.15%.

[1] Test Of English as a Foreign Language.
[2] International English Language Testing System.

Table 1
Number of phone tokens for different mispronunciation error types.

Error type	#Correct	#Substitution	#Deletion	#Insertion
Number	103,522	15,673	3180	1255

Table 2
Phone tokens for correct and incorrect pronunciations on different data sets.

	#Correct	#Incorrect	% Misp. rate
Training	74,798	11,184	13.01
validation	11,942	1846	13.39
Test	16,782	2643	13.61
Total	103,522	15,673	13.15

4.2. ASR performance of different acoustic models

Baseline acoustic model is firstly trained as context dependent GMM-HMM models (CD-GMM-HMM) in the Maximum Likelihood (ML) sense. All these native speech data are collected in 8-bit mu-law format, sampled in 8 kHz. The acoustic features, extracted by a 25 ms hamming window with a 10 ms time shift, consist of 13-dim MFCC and their first and second-order time derivatives. The cepstral mean normalization is performed for each utterance. Three-states, left-to-right HMM triphone models, each state with 16 Gaussian components, diagonal covariance output distribution, are adopted. The CMU pronunciation dictionary phone set with lexical stress, totally 69 different phones, is used for the acoustic model training.

Acoustic models are then enhanced by DNN training (Seide et al., 2011). Our baseline DNN model (MFCC-DNN-HMM) is a 5 layer network, consisting of 1 input layer, 3 hidden layers (each layer with 2 K units) and 1 output layer, with the same number of senones as that of CD-GMM-HMM. The input of DNN is an augmented feature vector, which contains 5 left frames, the current frame and 5 right frames. Each dimension is normalized to zero mean and unity variance.

This pitch embedded DNN model, PIT-DNN-HMM, has the same configurations as MFCC-DNN-HMM, except the input features, where 3 dimensional pitch related features are appended to MFCCs, i.e., raw log F0 and its dynamic counterparts (delta and delta delta) for voiced region, while for unvoiced region, three dimensions are all set to zeros. This pitch embedding method and training recipes of these CD-GMM-HMM and DNN models are almost the same as those used in our previous work (Hu et al., 2014a).

We observe experimentally that the ASR performance of native acoustic models is highly correlated with the performance of mispronunciation detection, therefore, we firstly assess the quality of those acoustic models. We evaluate their speech recognition performances on that held-out word test set. A silence-word-silence word-net or free

Table 3
Recognition performance of different acoustic models.

	WER (%)		PER (%)	
CD-GMM-HMM	6.95		38.75	
MFCC-DNN-HMM	3.00	(+56.8%)	25.43	(+34.4%)
PIT-DNN-HMM	2.80	(+59.7%)	24.70	(+36.3%)

phone loop is adopted for word level or phone level recognition performance evaluation, respectively. The Word Error Rate (WER) and Phone Error Rate (PER) are shown in Table 3. The calculation of WER is exactly the same as that of isolated word recognition, in which a word graph is built for the given vocabulary. A word with any occurred errors, including phone substitutions, deletions and insertions, are regarded as an erroneous word. Word deletions and insertions are not allowed due to the isolated, single word assumption in decoding each utterance. Compared with the baseline CD-GMM-HMM model, the relative improvements are shown in parentheses correspondingly. The MFCC-DNN-HMM model can outperform baseline GMM model by a large margin, and the performance can be further improved slightly by modeling F0 in DNN. This CD-GMM-HMM model and pitch embedded DNN model, PIT-DNN-HMM, will be used in the following mispronunciation detection experiments.

4.3. Performance metric for mispronunciation detection

To evaluate the performance of mispronunciation detection, precision, recall and the total accuracy are used in our experiments, which are defined as follows:

$$Precision = \frac{N_M}{N_D} \times 100\% \tag{16}$$

$$Recall = \frac{N_M}{N_H} \times 100\% \tag{17}$$

$$Accuracy = \frac{N_M + N_C}{N} \times 100\% \tag{18}$$

where N_M is the number of true mispronunciations detected by the system; N_D is the total number of mispronunciations detected by the system; N_H is the total number of mispronunciations judged by the human evaluators; N_C is the number of true correct pronunciations detected by the system and N is the total number of phone tokens.

4.4. GOP based approaches: from GMM to DNN

We have extended GOP based approach from GMM-HMM based system to DNN-HMM based system and further revised the GOP measure. Here, we compare the mispronunciation detection performance of those three systems, i.e., GOP-GMM, GOP-DNN1 and GOP-DNN2, on the non-native language learning corpus. A single threshold approach is used and the Precision–Recall curve, based on human label result on the test set, is drawn in Fig. 4. GOP-GMM stands for the GOP measure in GMM-HMM based system, which is defined as Eq. (4). GOP-DNN1 and GOP-DNN2 are two GOP measures in DNN-HMM based system, whose phone segmental posterior is defined as Eq. (6) or Eq. (7), respectively.

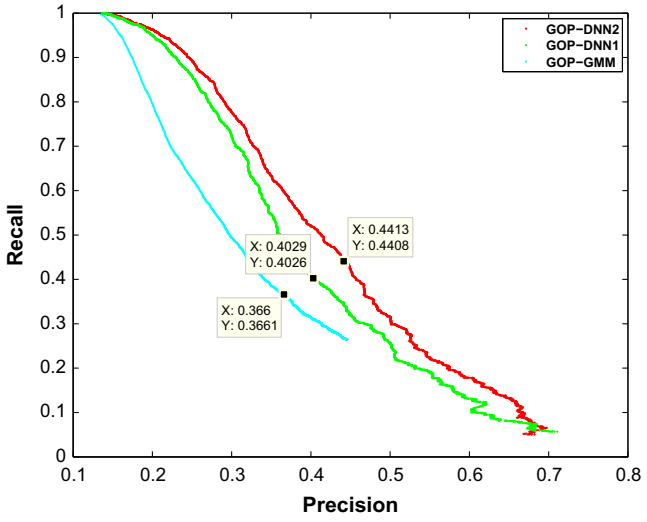


Fig. 4. Comparison of three GOP based systems.

As shown in the figure, two DNN-HMM based GOP systems outperform GMM-HMM based system consistently. The precision or recall rate, at the operating point where precision equals recall, is improved from 36.6% to 40.2% by comparing GOP-DNN1 with GOP-GMM system. The better performance mainly benefits from the improved discriminability of DNN models. This rate is further improved by 3.9% by using the revised GOP measure (GOP-DNN2).

In the following sections, GOP-DNN2 is adopted as the best GOP based system and set as the baseline when compared with classifier based systems.

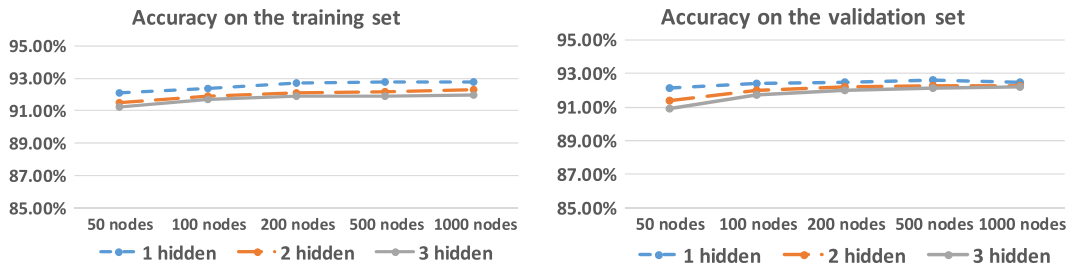
4.5. Classifier based approaches

SVM based classifier and our proposed NN-based LR classifier are built for mispronunciation detection.

4.5.1. Training of NN-based LR classifier

Different neural network topologies, e.g., different number of hidden layers and hidden nodes for each layer, are studied. The scoring accuracy on the training and validation sets are shown in Fig. 5 and Table 4, where the default threshold ($\theta = 0.5$) is used. The results show that detection accuracies of various topologies of neural network, i.e., different number of hidden layers and nodes per layer, do not change significantly on both the training set and validation set. The structure of 1 hidden layer with 500 nodes achieves the highest accuracy experimentally. Based on these observations on validation set, the model of 1 hidden layer with 500 nodes is used in our following experiments.

Following the conventional approach, we trained NN-based, 2-class LR classifier for each individual phone



(a) Training set (b) Validation set

Fig. 5. Results of different neural network topologies.

Table 4
Test accuracy on validation set for different neural networks (1 hidden layer with different number of hidden nodes).

hidden nodes	50	100	200	500	1000
Accuracy(%)	92.1	92.4	92.5	92.6	92.5

without using a common hidden layer, i.e., separately trained, and compare their performances with the proposed, NN-based LR classifier with shared hidden layer.

To illustrate the number of training tokens of individual phones and their effects on the classification performance, we choose the classifiers trained for 5 phones, '[ə]', '[iː]', '[e]', '[ɑː]' and '[ʒ]' for comparison. Among all these phones, '[ə]' has the most training tokens, '[ʒ]' has the least training tokens and the other three phones have a moderate size. The token prior, percentage of phone specific tokens in all the training set, is shown in Table 5. We vary the number of hidden nodes of each individually trained phone-dependent, NN-based, LR classifiers so as to obtain the best performance. The optimal, judging from the test result on validation set, separately trained classifiers with no shared hidden layer are then compared with the proposed, NN-based LR classifier with a shared hidden layer in their corresponding precision, recall and overall accuracy.

The results are depicted in Table 5. By sharing hidden layers across different phones, our proposed LR classifier outperforms the one individually trained without sharing. The most dramatic improvement of precision rate or recall rate is for the phone '[ʒ]' or '[ɑː]', respectively. The improvements of total accuracy of these two phones are also significant. These two phones share the same property that the size of its phone specific training tokens is relatively small. Therefore, we conclude that sharing data in training a common hidden layer can significantly alleviate the most serious training data shortage problem for those phones with less tokens. Slight improvement or equal mispronunciation detection performance is also observed for other phones by sharing hidden layers across different phones.

4.5.2. Comparison of different mispronunciation detection systems

We compare the detection performance of three systems, i.e., the best GOP, SVM and NN-based LR classifier, for individual phones. 15 most frequently mispronounced phones are chosen according to human labeled data. These 15 phones account for more than 85% of all mispronunciations. Table 6 gives the performance of these 3 methods for these 15 phones (in the order of its size of incorrect tokens) and all phones. The result is given at the operating point where precision rate equals the recall rate on the test set. The results show that SVM and NN-based LR classifiers outperform GOP based method consistently. This is because the classifier based methods use more features rather than one GOP score and are trained in a supervised mode. NN-based LR classifier yields a higher performance for most phones except for three phones, compared with SVM.

Fig. 6 further illustrates the overall mispronunciation detection performance for all the phones on the test set for these three systems. These precision–recall curves are drawn with a single threshold approach. As shown in the figure, classifier based methods significantly outperform GOP based method, e.g., more than 22% equal precision–recall rate improvement is obtained. The performance of GOP based approach can be improved by setting different thresholds for different phones. Compared with

Table 5
Comparison of two kinds of classifiers: NN-based LR classifier with a shared hidden layer and individually trained classifier with only phone specific data.

	Prior (%)	Jointly trained (%)			Individually trained (%)		
		Precision	Recall	Accuracy	Precision	Recall	Accuracy
'[ə]'	7.88	79.5	66.6	82.6	79.3	64.4	82
'[iː]'	3.02	83.8	43.1	92.6	79.2	43.1	92.2
'[e]'	2.51	88.0	83.1	89.6	89.0	82.4	89.8
'[ɑː]'	1.96	85.6	80.5	82.4	84.0	73.5	78.4
'[ʒ]'	0.1	76.9	100	80.0	66.7	100	66.7

Table 6
The mispronunciation detection performance for individual phones where precision is set the same as recall.

	Mis. rate (%)	GOP-based (%)	Classifier-based (%)	
		GOP-DNN2	SVM	LR
'[ə]'	34.4	64.1	73.0	74.6
'[z]'	54.7	64.1	73.8	76.6
'[ɑː]'	53.3	73.5	80.0	83.7
'[əː]'	32.0	53.1	59.4	63.5
'[e]'	36.8	67.2	84.5	86.7
'[æ]'	29.6	65.2	66.7	67.5
'[ei]'	24.7	48.2	52.4	52.7
'[əu]'	29.6	59.4	68.9	67.9
'[i]'	8.1	25.8	56.4	53.1
'[θ]'	54.0	64.0	69.6	72.0
'[iː]'	11.4	59.7	62.0	60.2
'[ɔː]'	19.6	45.3	51.6	56.3
'[v]'	24.4	68.3	71.7	73.3
'[ɑi]'	18.0	69.1	78.2	81.1
'[r]'	5.1	18.6	18.6	20.9
all	13.6	44.0	67.0	69.2

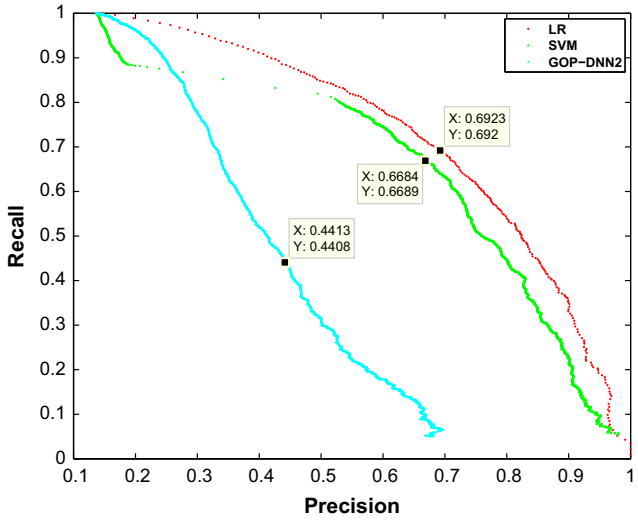


Fig. 6. Test performance of the best GOP, SVM and NN-based LR classifier for English mispronunciation detection.

conventional SVM approach, our proposed NN-based LR classifier not only streamlines training multiple individual classifiers separately, but also learns a better feature representation for improving mispronunciation detection performance, e.g., both the precision and recall are improved by 2.2% at the operating point where precision equals recall.

5. Mandarin mispronunciation detection

To evaluate the efficiency of our approaches to the mispronunciation detection task of other languages, we test them on a continuous read, L2 Mandarin language learning corpus.

5.1. A brief introduction of Mandarin Chinese

Mandarin Chinese, the official language in China, is the most widely used tonal language in terms of speaking population. Each Chinese character, which is a morpheme in written Chinese, is pronounced as a tonal syllable, i.e., a base syllable plus a lexical tone. All Mandarin syllables have a structure of (consonant)-vowel-(consonant), where only the vowel nucleus is an obligatory component. A mandarin syllable without tone label is referred as a base syllable and with tone label is referred as a tonal syllable. By the convention of Chinese phonology, each base syllable can be divided into two parts: initial and final. The initial (onset) includes what precedes the vowel while the final includes the vowel (nucleus) and what succeeds it (coda). Most Mandarin initials are unvoiced and the tones are carried primarily by the finals. For each vowel, there are 5 different tones, i.e., the 1st, 2st, 3st, 4st and neutral tone.

In this study, a tonal syllable is divided into two parts: initial and tonal final. e.g., for the tonal syllable 'ma1', its initial is 'm' and its tonal final is 'a1', where '1' standards for the lexical tone. Therefore, the phone set consists of initial phones and tonal final phones.

5.2. Databases

A Mandarin corpus, recorded by 110 native speakers (gender balanced) with standard pronunciations, is used to train the native acoustic model for our Mandarin CALL system, totaling ~41 h. The recording transcription includes single tonal syllables, multi-syllablic words and sentences. An extra data set, 30 speakers, ~6.5 h, is used to evaluate the ASR performance of trained acoustic models.

A large scale Mandarin learning corpus, iCALL corpus (Chen et al., 2013), is used to evaluate the performance of mispronunciation detection. This corpus is recorded by 300 beginner learners of Mandarin Chinese, whose mother tongues are mainly European origin, i.e., Germanic, Romance and Slavic. The speech data are transcribed in Pinyin through perceptual listening tasks by 3 native linguistic experts. By comparing the labeled surface pronunciation and canonical pronunciation prompts, we get the mispronunciation type for each utterance on phone level (initial or tonal final). There are more than 20k labeled utterances in total. 15k utterances, randomly selected, are used for training and the remaining 5k utterances are used to evaluate the performance of different detection approaches.

5.3. Performance for different systems

Being the same as acoustic model training in English mispronunciation detection systems, we first train a context dependent GMM-HMM acoustic model and then enhance its discriminability by DNN training and pitch embedding. Mandarin Chinese is a kind of tonal language, where tone plays an important role to distinguish different tonal finals.

Table 7
The Mandarin mispronunciation detection performance for individual phones where precision is set the same as recall.

	Misp. rate (%)	GOP-based (%)			Classifier-based (%)	
		GOP-GMM	GOP-DNN1	GOP-DNN2	SVM	LR
an4	46.5	61.1	80.2	83.8	84.5	86.3
u4	36.4	47.8	73.4	76.6	79.4	79.7
i4	42.6	56.0	71.5	72.1	82.1	81.5
e4	40.1	52.0	68.2	71.8	78.7	78.7
an1	39.3	46.5	67.4	69.9	71.9	75.3
i3	47.0	55.9	63.7	63.7	65.4	71.2
an2	37.0	45.5	70.1	70.6	77.7	78.8
an3	58.6	68.5	69.6	70.1	71.9	77.4
ao3	42.8	51.1	59.5	61.3	62.7	68.3
ang4	50.5	60.1	79.4	82.4	88.6	89.7
a1	32.0	36.7	56.2	58.8	74.1	76.1
i1	41.5	48.3	62.5	64.8	68.0	69.5
ao4	36.9	50.4	72.4	73.4	82.8	82.4
a4	37.3	50.5	64.1	68.6	79.8	81.4
e	18.6	25.4	33.6	34.9	36.5	43.2
zh	10.8	37.4	48.5	53.4	53.7	54.5
sh	7.4	25.3	44.1	44.3	59.3	61.4
z	12.5	36.9	52.1	58.8	54.3	60.6
t	10.0	37.2	50.6	58.3	63.0	61.7
j	11.0	34.2	48.8	54.2	52.4	57.3
all	24.1	37.0	60.0	62.8	71.9	75.5

Therefore, modeling F0, the main acoustic cue of tone, in acoustic model training will significantly improve the performance for both ASR and mispronunciation detection.

Three GOP based systems, i.e., GOP-GMM, GOP-DNN1 and GOP-DNN2, and two classifier based systems, i.e., SVM and NN-based LR classifier, are built. The detection performance of different systems for individual phones are illustrated in Table 7, which shows the mispronunciation rate and detection performance of 15 most frequently mispronounced tonal finals and 5 most frequently mispronounced initials according to human labeled data. The mispronunciation rate is shown in the second column. The mispronunciation detection performance is given at the operating point where precision rate equals recall rate on the test set. For those three GOP based systems, the mispronunciation detection performance is improved successively, consistently for different phones, by comparing the baseline GOP-GMM (GMM-HMM based system) with GOP-DNN1 (pitch embedded, DNN-HMM based system) and GOP-DNN2 (applying the promoted GOP measure). Classifier based systems achieve higher detection performance than GOP based systems for all the phones. NN-based LR classifier yields a higher precision or recall rate than SVM for most phones. These observations are almost the same as what we observed in English mispronunciation detection systems. The overall performances (considering all the initial and tonal finals on the test set) of the best GOP approach, SVM and our proposed NN-based LR classifier are shown in Fig. 7. A single threshold approach is adopted. It shows that our proposed NN-based LR classifier can outperform the SVM classifier by 3.4% and GOP-DNN2 by 12.8% of equal precision–recall rate improvement.

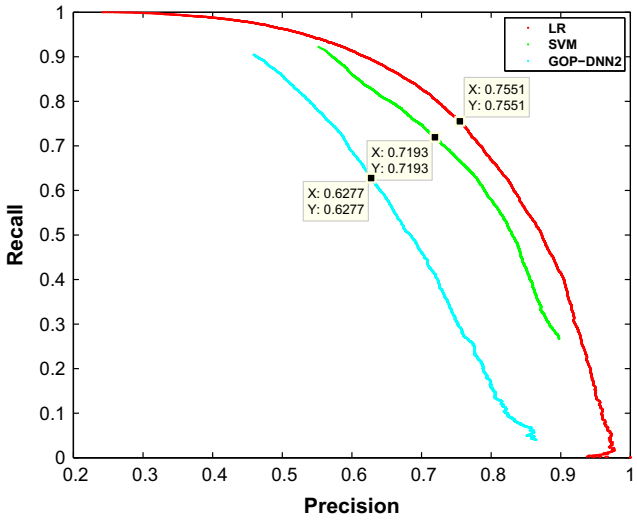


Fig. 7. Test performance of the best GOP, SVM and NN-based LR classifier for Mandarin mispronunciation detection.

6. Realtime system implementation

Using DNN based acoustic model to calculate GOP scores has a high computational complexity due to the calculations in forward propagation of deep structure. To build an online real time mispronunciation detection system based on DNN trained acoustic model, it will have a much higher latency than the conventional GMM-HMM based systems. To speed up the processing of realtime system, two acceleration methods, restructuring of DNN models by Singular Value Decomposition (SVD) (Xue

et al., 2013) and frame asynchronous decoding (Vanhoucke et al., 2013), are employed to our online system. We reconstruct DNN models by applying SVD on the inherent sparse weight matrices of DNN. This restructured model is then fine-tuned with back propagation method to get back its modeling accuracy. In our realtime system, the original DNN model is configured as three hidden layer and 2k nodes for each layer. By keeping 40% of accumulated rank-ordered singular values for each layer, DNN model size can be reduced by 67.2% without any degradation of mispronunciation detection performance. Neural networks can be able to learn from very long observation window going up to 400 ms. Given this very long temporal context, running neural network at a lower frame rate rather than the typical 10 ms can achieve the comparable performance. In our system, the decoding is run frame-asynchronously at a half of frame rate of feature stream by simply duplicating the previous frame's predictions every even number of frame. With this approach, we can further reduce 50% of the computational cost with marginal mispronunciation detection performance reduction. A cloud computing (Microsoft Azure) based realtime system can be tried through (http://bing.msn.cn/dict/).

7. Conclusions

In this paper, we propose several approaches to mispronunciation detection improvement. The GMM-HMM based acoustic model, which is used to calculate GOP, is first refined by DNN training to enhance its discrimination. F0 is also added to DNN based model to detect the pronunciation errors caused by misusing lexical stress or lexical tone for L2 language learners; then the GOP measure in DNN-HMM based system is improved to robustly evaluate the pronunciation quality of non-native learners by native speakers' acoustic models; finally, a neural network based logistic regression classifier is proposed to by-pass the effort in training multiple individual classifiers separately, and further improve the generalization capability. The experimental results on both English and Chinese learning corpora show the efficiencies of our proposed approaches. In the future, we will investigate on mispronunciation detection of insertion, deletion and prosodic errors so as to provide more informative diagnosis feedbacks to the language learners.

References

Bahl, L.R., Brown, P.F., de Souza, P.V., Mercer, R.L., 1986. Maximum mutual information estimation of hidden Markov model parameters for speech recognition. Proc. ICASSP-1986. IEEE, pp. 49–52.

Bengio, Y., Courville, A., Vincent, P., 2013. Representation learning: a review and new perspectives. IEEE Trans. Pattern Anal. Mach. Intell. 35, 1798–1828.

Bishop, C.M., 2006. Pattern Recognition and Machine Learning (Information Science and Statistics). Springer-Verlag, New York, Inc., Secaucus, NJ, USA.

Chen, N.F., Shivakumar, V., Harikumar, M., Ma, B., Li, H., 2013. Large-scale characterization of Mandarin pronunciation errors made by native speakers of European languages. Proc. InterSpeech-2013. ISCA, pp. 2370–2374.

Dahl, G.E., Yu, D., Deng, L., Acero, A., 2012. Context-dependent pre-trained deep neural networks for large-vocabulary speech recognition. IEEE Trans. Audio, Speech Lang. Process. 20, 30–42.

Franco, H., Neumeyer, L., Ramos, M., Bratt, H., 1999. Automatic detection of phone-level mispronunciation for language learning. Proc. Eurospeech-1999. ISCA, pp. 851–854.

Hinton, G.E., Osindero, S., Teh, Y.W., 2006. A fast learning algorithm for deep belief nets. Neural Comput. 18, 1527–1554.

Hinton, G.E., Deng, L., Yu, D., Dahl, G.E., Rahman Mohamed, A., Jaitly, N., Senior, A., Vanhoucke, V., Nguyen, P., Sainath, T.N., Kingsbury, B., 2012a. Deep neural networks for acoustic modeling in speech recognition. IEEE Signal Process. Mag. 29 (6), 82–97.

Hinton, G.E., Srivastava, N., Krizhevsky, A., Sutskever, I., Salakhutdinov, R., 2012b. Improving Neural Networks by Preventing Co-adaptation of Feature Detectors. Tech. Rep. 1207.0580.

Hirabayashi, K., Nakagawa, S., 2010. Automatic evaluation of English pronunciation by Japanese speakers using various acoustic features and pattern recognition techniques. Proc. InterSpeech-2010. ISCA, pp. 598–601.

Huang, J.-T., Li, J., Yu, D., Deng, L., Gong, Y., 2013. Cross-language knowledge transfer using multilingual deep neural network with shared hidden layers. Proc. ICASSP-2013. IEEE, pp. 7304–7308.

Hu, W., Qian, Y., Soong, F.K., 2013. A new DNN-based high quality pronunciation evaluation for computer-aided language learning (CALL). Proc. InterSpeech-2013. ISCA, pp. 1886–1890.

Hu, W., Qian, Y., Soong, F.K., 2014a. A DNN-based acoustic modeling of tonal language and its application to Mandarin pronunciation training. Proc. ICASSP-2014. IEEE, pp. 3230–3234.

Hu, W., Qian, Y., Soong, F.K., 2014b. A new neural network based logistic regression classifier for improving mispronunciation detection of L2 language learners. Proc. ISCSLP-2014. IEEE, pp. 245–249.

Ito, A., Lim, Y.-L., Suzuki, M., Makino, S., 2005. Pronunciation error detection method based on error rule clustering using a decision tree. Proc. Eurospeech-2005. ISCA, pp. 173–176.

Jiang, H., 2005. Confidence measures for speech recognition: a survey. Speech Commun. 45 (4), 455–470.

Jie, J., Xu, B., 2009. Exploring the automatic mispronunciation detection of confusable phones for Mandarin. Proc. ICASSP-2009. IEEE, pp. 4833–4836.

Joachims, T., 1998. Making Large-scale SVM Learning Practical. LS8-Report 24, Universität Dortmund, LS VIII-Report.

Juang, B.H., Chou, W., Lee, C.H., 1997. Minimum classification error rate methods for speech recognition. IEEE Trans. Speech Audio Process. 5 (3), 266–277.

Kim, Y., Franco, H., Neumeyer, L., 1997. Automatic pronunciation scoring of specific phone segments for language instruction. Proc. Eurospeech-1997. ISCA, pp. 645–648.

Pitrelli, J.F., Fong, C., 1995. Phonebook: NYNEX Isolated Words Linguistic Data Consortium, Philadelphia. <http://catalog.ldc.upenn.edu/LDC95S27>.

Povey, D., Woodland, P.C., 2002. Minimum phone error and i-smoothing for improved discriminative training. Proc. ICASSP-2002. IEEE, pp. 105–108.

Qian, X., Soong, F.K., Meng, H.M., 2010. Discriminative acoustic model for improving mispronunciation detection and diagnosis in computer-aided pronunciation training (CAPT). Proc. InterSpeech-2010. ISCA, pp. 757–760.

Qian, X., Meng, H.M., Soong, F.K., 2012. The use of DBN-HMMs for mispronunciation detection and diagnosis in L2 English to support computer-aided pronunciation training. Proc. InterSpeech-2012. ISCA.

Rumelhart, D.E., Hinton, G.E., Williams, R.J., 1986. Learning representations by back-propagating errors. Nature 323, 533–536.

Seide, F., Li, G., Chen, X., Yu, D., 2011. Feature engineering in context-dependent deep neural networks for conversational speech transcription. Proc. ASRU-2011. IEEE, pp. 24–29.

Soong, F.K., Kit Lo, W., Nakamura, S., 2004. Generalized word posterior probability (GWPP) for measuring reliability of recognized words. In: Proc. SWIM-2004.

Strik, H., Truong, K., de Wet, F., Cucchiarini, C., 2009. Comparing different approaches for automatic pronunciation error detection. Speech Commun. 51 (10), 845–852.

Tokuda, K., Mausko, T., Miyazaki, N., Kobayashi, T., 2002. Multi-space probability distribution HMM. IEICE Trans. Inform. Syst., 455–464

Truong, K., 2004. Automatic Pronunciation Error Detection in Dutch as a Second Language: An Acoustic–phonetic Approach. Master's Thesis, Utrecht University, The Netherlands.

van Doremalen, J., Cucchiarini, C., Strik, H., 2009. Automatic detection of vowel pronunciation errors using multiple information sources. Proc. ASRU-2009. IEEE, pp. 580–585.

Vanhoucke, V., Devin, M., Heigold, G., 2013. Multiframe deep neural networks for acoustic modeling. Proc. ICASSP-2013. IEEE, pp. 7582–7585.

Vapnik, V.N., 1995. The Nature of Statistical Learning Theory. Springer.

Wang, Y.-B., Lee, L.-S., 2012. Improved approaches of modeling and detecting error patterns with empirical analysis for computer-aided pronunciation training. Proc. ICASSP-2012. IEEE, pp. 5049–5052.

Wei, S., Hu, G., Hu, Y., Wang, R.H., 2009. A new method for mispronunciation detection using support vector machine based on pronunciation space models. Speech Commun. 51 (10), 896–905.

Witt, S.M., 2012. Automatic error detection in pronunciation training:where we are and where we need to go. Proc. Int. Sym. on Automatic Detection of Errors in Pronunciation Training 2012. ISCA, pp. 1–8.

Witt, S.M., Young, S.J., 2000. Phone-level pronunciation scoring and assessment for interactive language learning. Speech Commun. 30 (2–3), 95–108.

Xue, J., Li, J., Gong, Y., 2013. Restructuring of deep neural network acoustic models with singular value decomposition. Proc. InterSpeech-2013. ISCA, pp. 2365–2369.

Xu, S., Jiang, J., Chen, Z., Xu, B., 2009. Automatic pronunciation error detection based on linguistic knowledge and pronunciation space. Proc. ICASSP-2009. IEEE, pp. 4841–4844.

Yan, K., Gong, S., 2011. Pronunciation proficiency evaluation based on discriminatively refined acoustic models. IJITCS 3, 17–23.

Yoon, S.-Y., Hasegawa-Johnson, M., Sproat, R., 2010. Landmark-based automated pronunciation error detection. Proc. InterSpeech-2010. ISCA, pp. 614–617.

Zhang, F., Huang, C., Soong, F.K., Chu, M., Wang, R.H., 2008. Automatic mispronunciation detection for Mandarin. Proc. ICASSP-2008. IEEE, pp. 5077–5080.