Unsupervised Pronunciation Fluency Scoring by infoGan

Wenwei Dong*, Yanlu Xie* and Binghuai Lin†
* Beijing Advanced Innovation Center for Language Resources, Beijing Language and Culture University, Beijing, China
E-mail: dongwenwei_blcu@163.com
E-mail: xieyanlu@blcu.edu.cn
† MIG, Tencent Science and Technology Ltd., Beijing, China
E-mail: binghuailin@tencent.com

***Abstract*—Pronunciation fluency scoring (PFS) is a primary task in computer-aided second language (L2) learning. Most of existing PFS algorithms are based on supervised learning, where human-labeled scores are used to train the scoring model. However, the human labeling is rather costly and tends to be biased. In order to tackle this problem, we propose an unsupervised learning approach, where an infoGan model is constructed to infer latent speech codes, and then these codes are used to build a classifier that distinguishes native and foreign speech. We found that this native-foreign classifier can generate good utterance-based fluency scores.**

I. Introduction

Language learning become more and more popular nowdays. A lot of language learners want to practice their pronunciation and test their oral English level. Computer-Assisted Pronunciation Training (CAPT) become important when huaman scoring is subjective and the feedback is not timely. In terms of feedbacks, it can be classified into two categories: mispronunciation type feedback can indicate specific error types, score feedback can indicate goodness of pronunciation. mispronunciation type feedback is mostly used in pronunciation training, while score feedback is more used in pronunciation evaluation task.

CAPT still has following difficulties: non-native speech data is hard to collect and annotate. Neural network needs lots of data to train but collecting and annotating data will cost a lot. The consistency of manual socring between humans is unsatisfactory. Because of data limitation, pronunciation scoring of early stage mainly based on template [1]. Teacher and student read same scripts, and teacher's speech feature is used as template to compute the distance with student acoustic feature. Truong [2] used a binary classifier to distinguish confusing phone pairs. Lee [3], under the condition of low resources, compared MFCC, gaussian posteriorgrams and English phoneme state posteriorgrams ,and found that transfering knowledge from rich resources language can benfit scoring task. Kyriakopoulos [4] used siamese phone distance feature with attention mechanism to predict scores. some of researches [5-7] used ASR frameworks to scoring. Native data are easier to collect than non-native, so they use native speech as training data to score non-native pronunciation, but there are some mismatchs in channel and speaker and other factors. Some work [8-9] tried to use speaker normalization to reduce the influence of mismatch. those methods are text denpendence.

Since the supervised learning methods need a lot of labeled data and the templated methods need same speech contents from both teacher and student. Some researchers tried to use unsupervised learning and parameter sharing methods to reduce the cost of annotation. Wang [10] and Miao [11] used DNN posterior as feature to unsupervised cluster, they tried to discover mispronunciation pattern of second language learners and replaced human annotation. [12] used a subspace Gaussian mixture model trained with both phonetic and prosodic features to predict human rated fluency scores on read spoken by non-native learners of Mandarin.

Most of the methods scored segments and supersegments separately. We intend to score non-native speech by using its similarity with native speech. The more scores distribution of non-native speech similar to the native speech, the higher the score, it's a text independence method. GAN as one of unsupervised model has successfully used in many domains. InfoGan [13] as a kind of GAN variants, it decomposed GAN's unstructured noise vector into two parts: incompressible noise and latent code which can learn the latent data distribution. InfoGan has been successfully used in image recognition. We proposed to use infoGan as feature extractor to learn the native and non-native data distribution, then use classifier to predict pronunciation scores. We also compare two mapping methods of frame level score to utterance level: the mean of each frame and native-templated Jensen-Shannon (JS) distance.

The paper organized as fellows: section 2 will introduce infoGan and softmax scoring framework, section 3 will intrduce experiment corpora and setups, results and discussion will be in section 4, and conclusion will be section 5.

II. InfoGan scoring framework

This section will introduce infoGan and different methods of combining frame level score to utterence and speaker level.

A. InfoGan

The GAN formulation uses a simple factored continuous input noise vector z, while imposing no restrictions on the manner in which the generator may use this noise. As a result, it is possible that the noise will be used by the generator in a highly entangled way, causing the individual dimensions of z to not correspond to semantic features of the data. Rather than using a single unstructured noise vector, infoGan decompose the input noise vector into two parts: (i) z, which is treated as source of incompressible noise; (ii) c, which called the latent code and targeted the salient structured semantic features of the data distribution. So we will provide the generator both noise and latent code c, but in standard GAN, the generator is free to ignore the additional latent code c by finding a solution satisfying $PG(x|c)$ = PG(x). So infoGan propose an information-theoretic regularization: there should be high mutual information between latent codes c and generator distribution G(z, c). Thus I(c;G(z, c)) should be high. The mutual information term I(c;G(z, c)) is hard to maximize directly as it requires access to the posterior $P(c|x)$. Then infoGan obtain a lower bound of it by defining an auxiliary distribution $Q(c|x)$ to approximate $P(c|x)$, so it add a classifier Q to achieving training objectives. InfoGAN is defined as the following minimax game with a variational regularization of mutual information and a hyperparameter λ:

$$\underset{G,Q}{min}\,\underset{D}{max}\, V_{InfoGAN}(D,G,Q) = V(D,G) - \lambda L_I(G,Q) \quad (1)$$

where

$$\underset{G}{min}\,\underset{D}{max}\, V(D,G) = \underset{x\sim p_{data}}{\mathbb{E}}[logD(x)] + \underset{z\sim noise}{\mathbb{E}}[log(1-D(G(z)))] \quad (2)$$

It disentangle both discrete and continuous latent factors, scale to complicated datasets, and typically requires no more training time than regular GAN, because Q and D share the hidden layers.

B. Scoring Methods

We have used two ways to score utterences: the first is using the mean of each frame scores in an utterence as utterence score and using the mean of each utterence as speaker score. Second is using the JS distance as scores.

$$JS(P\|Q) = \frac{1}{2}KL(P(x)\|\frac{P(x)+Q(x)}{2}) + \frac{1}{2}KL(Q(x)\|\frac{P(x)+Q(x)}{2}) \quad (3)$$

where KL($P\|Q$)

$$KL(P\|Q) = \sum P(x)log\frac{P(x)}{Q(x)}) \quad (4)$$

Kulback-Leibler (KL) distance, also known as relative entropy, is a method of discribing the difference between two probability distributions P and Q. It is asymmetrical, which means $KL(P\|Q) \neq KL(Q\|P)$.

JS distance measures the similarity of two distributions, it's a variant of KL distance and solves the problem of asymmetry of KL distance. So JS distance is symmetrical, and its value is between 0 and 1. In non-native dataset. We count the number of frame scores at 0-10 and 10-20 and so on at utterence and speaker level. We can get a 10-dimsional vector that discribing the number of frames per score level, then it is normalized to 0-1 through dividing it by total number of frames of an utterence or speaker. In the native dataset, we count the frames of whole dataset to form a template vector, so the vector pairs from native and non-native can be used to calculate JS distance.

C. Framework Detail

We try two ways of using infoGan. The one is using Q as classifier and the other is using Q as feature extractor. The framework is shown in Fig. 1.

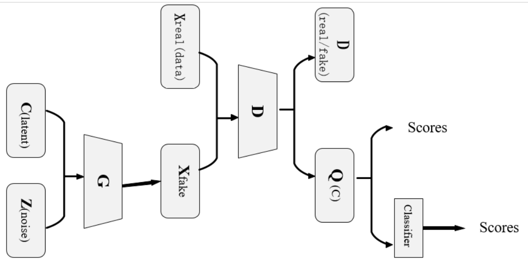

Fig. 1. InfoGan scoring framework.

In infoGan, Q's ouptut size are equal to latent code size c. We test different latent code size c and let it learn different information. We use both native and non-native data as training set. The discriminator discriminate real or fake data. The Q classifier can classify native or non-native data. The confidence score that be classified as native is the score of each frame. In the training process, labels can be generated randomly. The model can learn the relationship between features and labels.

III. Experiment Corpora and Setups

A. Speech Corpora

The corpora used in study consist of four parts. Training sets consist of 10 hours WSJ as native corpus and 10 hours Chinese speaking English (CSE) dataset as non-native corpus. The testing set are TIMIT which is native English corpus, and ERJ [14] which is Jpanese speaking English corpus. the ERJ corpus has 2000 sentences and timit has about 1.7 hours. The above duration does not include slient segment. The ERJ corpus has annotated with utterence and speaker level scores, it contains 190 speakers. The Pearson correlation coefficient of utterence scores between humans is 0.55 and of speaker level is 0.796.

B. Expriment Setup

We use Kaldi toolkit to get 40-dimensional Fbank feature, and use Cepstral Mean and Variance Normalization (CMVN) method to reduce speaker difference, and VAD method to remove silent segment. The infoGan model have three parts. the generator have 4 layers, two dense layers have 1024 and 128 nodes respectively and two deconvolution layers. The discriminator and Q share the hidden layers, it consists of one convolution layer and one dense layer which have 64 and 128 nodes respectively. Q has softmax output layer which same size with latent code c and can be changed, D has 1 output layer with 2 nodes.

C. Evaluation Metric

Pearson correlation coefficient, also known as Pearson product-moment correlation coefficient (PPMCC or PCCs), is used to measure the correlation between two variables X and Y (linear correlation), whose value is between -1 and 1.

1) When the correlation coefficient is 0, the X and Y variables have no relationship.
2) When the vlaue of X increases (decreases), and the value of Y increases (decreases). the two variables are positively correlated and the correlation coefficient is between 0.00 and 1.00.
3) When the vlaue of X increases (decreases), and the value of Y decreases (increases., the two variables are negatively correlated and the correlation coefficient is between -1.00 and 0.00.

$$\rho_{X,Y} = \frac{cov(X,Y)}{\sigma_X \sigma_Y} = \frac{E[(X-\mu_X)(Y-\mu_Y)]}{\sigma_X \sigma_Y}) \qquad (5)$$

cov(X,Y) is covariance of variables X and Y, σ is the standard deviation.

IV. RESULTS AND DISCUSSION

First, we use Q as a binary classfier. The input of infoGan are WSJ and CSE data and no label is given. We want to figure out that whether infoGan can learn the distribution of two dataset through mutual information or not.

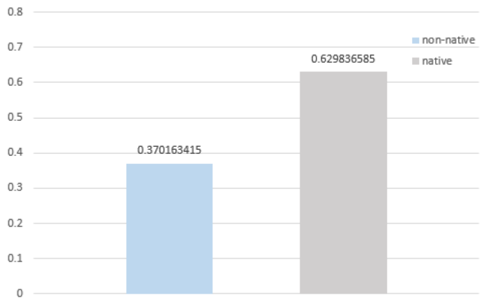

Fig. 2. Unsupervised classification result of timit.

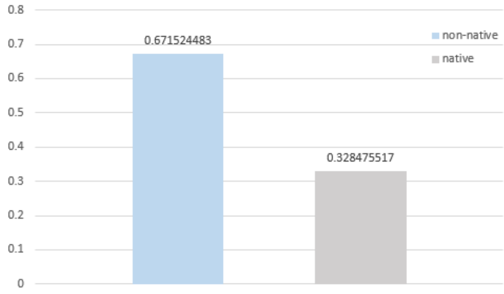

Fig. 3. Unsupervised classification result of ERJ.

Fig. 2. shows the testing result of ERJ. Fig. 3. shows the testing result of timit. From above figures, we know that the infoGan can learn the distribution of different dataset, but what kind of distributions still need to be futher studied, it can be channel or other environmental factors. So we use timit instead of WSJ to test model. Most of ERJ frames can be classified into non-native class and most of timit frames are classfied into native class. The model can classify the timit and ERJ with a totally unsupervised way. However, using Q as a classfier, the scoring result are not accurate enough. We want to further explore the Q can generate a better feature embedding.

We use Q as feature extractor, and use those feature softmax binary classifier to discriminate native or non-native speech. The output of it is two-dimensional posterior probability, those are the probability of the frame being classified as native or non-native. We use the native dimension probability as score of a frame. we static the number of score at different level. The results are shown in Fig. 4. and Fig. 5.

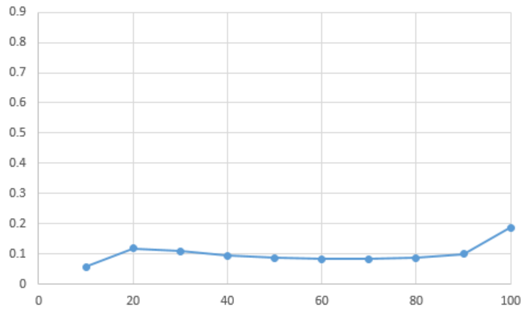

Fig. 4. Score distribution of ERJ.

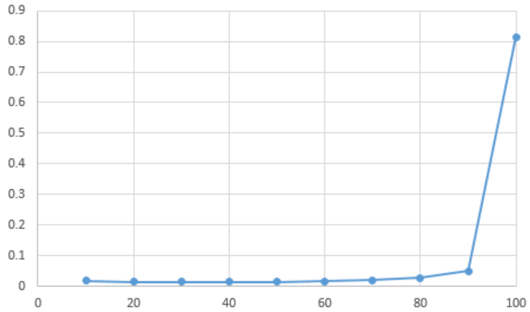

Fig. 5. Score distribution of timit.

In the Fig. 4. and Fig. 5, x-axis means scores in 0 to 100. The y-axis means the number of frames in each scores level and total number of frames ratio. From the figures we can see the native speech's score are more concentrated in high segemnts, and the score of non-native speech are more like uniform distribution. This is also in line with our understanding, the non-native's pronunciation level are uneven.

We set latent coede c size 10, then we further explore the different output size can impact the scoring or not, we try utterence level score and speaker level score.

We use UTT_MEAN, UTT_JS to denote uttenerce level methods: mean and JS distance, similarly, SPK_MEAN, SPK_JS denote speaker level mean and JS diatance methods.

The table 1 shows different latent code size. We use it as binary classifier input feature, as the size increase, the PCCs between model score and human score are higher, c = 10 get the best performance. At utterence level, the mean of each frame's score have more higher PCCs than JS distance. This because the number of frames in a sentence is too small to describe the distribution, so the mean of each frame scores perform better. At speaker level, the JS distance have higher PCCs than the method of taking the mean. When we set c = 2, we compare the infoGan and infoGan + softmax methods, the infogan+softmax methods perform better both in utterence and speaker level.

TABLE I
EXPERIMENT RESULTS

C	UTT_MEAN	UTT_JS	SPK_MEAN	SPK_JS	MODEL
2	0.036	0.031	0.031	0.074	infoGan
2	0.059	0.025	0.025	0.096	infoGan+softmax
3	0.057	0.016	0.019	0.122	infoGan+softmax
5	0.051	0.081	0.168	0.223	infoGan+softmax
7	0.137	0.099	0.161	0.232	infoGan+softmax
10	**0.201**	0.157	0.274	**0.310**	infoGan+softmax
12	0.050	0.060	0.107	0.121	infoGan+softmax
14	0.068	0.121	0.214	0.245	infoGan+softmax

V. CONCLUSION

We use infoGan to score non-native speech at both utterance and speaker level. First is using Q as binary classifier, it can learn the native and non-native data distribution on a unsupervised way, second is using Q as feature extractor, then use softmax to score,before the latent code c it shows as the Q feature dimension increase the correlation the human score are higher. The infoGan+softmax performs better than infoGan method. In utterence level scoring, the mean of frames score are better than JS distance, it may because the number of frames in a utterence are too small to describe a distribution. But in speaker level, each speaker have more frames to describe distribution, so the JS distance are perform better than mean mtehod.

ACKNOWLEDGMENT

This work is supported by National social Science foundation of China (18BYY124), Advanced Innovation Center for Language Resource and Intelligence (KYR17005), the Fundamental Research Funds for the Central Universities (18YJ030004), the Graduate Innovation Fund of Beijing Language and Culture University (19YCX130), and the project of "Intelligent Speech technology International Exchange". Yanlu Xie is the corresponding author.

REFERENCES

[1] S. .M. Goddijin,G. .D. Krom, "Evaluation of second language learners pronunciation using hidden Markov models," *Fifth European Conference on Speech Communication and Technology*. 1997.

[2] K. Truong, "Automatic pronunciation error detection in Dutch as a second language:an acoustic-phonetic approach." 2004.

[3] A. Lee and J. Glass, "Pronunciation assessment via a comparison-based system," in *Speech and Language Technology in Education*. 2013.

[4] K.. Kyriakopoulos,K. .M. Knill and ,M. .J. Gales, "A deep learning approach to assessing non-native pronunciation of English using phone distance," in *Proceedings of the Annual Conference of the International Speech Communication Association.,*Vol.2018, pp. 1626-1630.

[5] W. Hu, Y. Qian and ,F. .K. Soong, et al."Improved mispronunciation detection with deep neural network trained acoustic models and transfer learning based logistic regression classifiers," *Speech Communication.,* 2015.

[6] H. Huang, H. Xu and Y. Hu, et al. "A transfer learning approach to goodness of pronunciation based automatic mispronunciation detection," *Journal of the Acoustical Society of America,* 2017.

[7] W. Li, S. .M. Siniscalchi, N. .F. Chen and ,C. Lee, "Improving Non-Native Mispronunciation Detection and Enriching Diagnostic Feedback with DNN-Based Speech Attribute Modeling," *i*n ICASSP. 2016.

[8] D. Luo, C. Zhang, L. Xia and ,L. Wang, "Factorized Deep Neural Network Adaptation for Automatic Scoring of L2 Speech in English Speaking Tests," *i*n INTERSPEECH. 2018.

[9] G. Huang ,J. Ye, Z. Sun and ,Y. Shen, et al. "English mispronunciation detection based on improved GOP methods for Chinese students," *2017 International Conference on Progress in Informatics and Computing (PIC).* 2017.

[10] Y, B. Wang, L. .S. Lee, "Toward unsupervised discovery of pronunciation error patterns using universal phoneme posteriorgram for computer-assisted language learning," *i*n 2013 IEEE International Conference on Acoustictics, Speech and Signal Processing. pp. 8232-8236. 2013.

[11] S. Miao, X. Li, K. Li, et al. "Unsupervised Discovery of an Extended Phoneme Set in L2 English Speech for Mispronunciation Detection and Diagnosis," *i*n ICASSP. 2018.

[12] R. Tong, B. .P. Lim, N. .F. Chen, B. Ma and ,H. Li, "Subspace Guaussian Mixture Model for Computer-Assisted Language Learning," *i*n ICASSP. 2014.

[13] X. Chen, Y. Duan, R. Houthooft et al. "Infogan: Interpretable representation learning by information maximizing generative adversarial nets," *i*n Advances in neural information processing systems. 2016.

[14] N. Minematsu, Y. Tomiyama, K. Yoshimoto et al., "Development of English speech database read by Japanese to support CALL research," *i*n Proc. ICA. vol. 1, pp. 557-560. 2004.