Backend Design

---- Group 5: Yoyo D., Jiatong S., Nan H., Gary X., Rafa de la TO

Our backend model is a Computer-Assisted Language Learning model that takes audio/text as input and generates pronunciation scores for the sentences. For the spoken evaluation purpose, we define three goals for our backend system:

- Accurately detect pronunciation errors.
- High-grained score utterances, which can separate speakers between true and false pronunciation
- support for children

Following the purposes, we have three stages for the backend framework, including the acoustic model, the decoding model, and the scoring model.

Acoustic Model

The acoustic model is trained for phoneme recognization and their confidence level on pronunciation. We choose the state-of-the-art model (subsampling Factorized Time Delay Neural Network with Lattice-Free Maximum Mutual Information criterion - F-Chain-TDNN) supported by Kaldi for the model structure and applied the Hidden Markov Model for phonetic transitions. The training data we selected is Librispeech, a popular open-source corpus with 1000 hours of speech. Since the speakers of the Librispeech are all native speakers, the model could have less probability to learn mispronunciation errors. Since we did not have phoneme-level annotation for the corpus, we tune the model based on an Automatic Speech Recognition (ASR) tasks and employ the pre-built language model for decoding rescoring. For the ASR training and evaluation, we follow the official standard, which is using 960 hours for training, 20 hours for validation, and 20 hours for testing (10 hours clean test set and 10 hours noisy test set).

The Mel-frequency Cepstrum Coefficients (MFCC) are extracted for input features. For training, we follow the conventional process by incrementally feeding the data. Firstly, we start with 2,000 shortest utterances from a clean subset of the Librispeech and train a mono-phone system for initial alignments. Next, we gradually add data and feature transformation techniques (e.g. Linear Discriminant Analysis, Maximum Likelihood Linear Transformation, and feature-space Maximum Likelihood Linear Regression) to train a base GMM-HMM (Gaussian Mixture model - Hidden Markov Model) model. We then employ the alignment of this base model to generate the samples for TDNN training. In addition to the MFCC feature, i-vector features are also extracted to represent speaker information in the TDNN training. The F-TDNN model has 17 Factorized TDNN layers besides an LDA affine layer next to the input and a linear layer next to the output. For each of the F-TDNN layers, we have a linear layer (160 units) for factorization, a central layer (1536 units) and a scaling layer. Multi-task learning is implemented with two losses, including the LF-MMI loss and the Cross-Entropy Loss.

Decoding Model

We do not follow the decoding process of the ASR. Instead, we apply force-alignment for the speech based on the phoneme posteriorgram and the reference phoneme sequence. The posteriorgram is obtained from the acoustic TDNN model and the phoneme sequence is

generated via CMUdict that includes word-to-phoneme references. We utilize Dynamic Time Warping to solve the alignment. However, we do not use linear alignment here, because it cannot solve the optional silence between phonemes. To tackle the problem, we generate a decoding graph regarding the optional silences between phonemes and use the Viterbi algorithm to find the best path.

Scoring Model

The Goodness of Pronunciation (GOP) is adopted for the baseline of our model. The definition of the GOP is as follows:

$$GOP(p) = \frac{log(P(p|O))}{L(O)} = \frac{log(\frac{P(O|p)P(p)}{\sum_{q \in Q} P(O|q)P(q)})}{L(O)}$$

where p is specific phoneme, O is the related audio sample to p, and q is all other phonemes. The GOP score cannot be applied for scoring directly since the range of it varies on each phoneme. Therefore, the GOP is always interpreted as binary classes to determine whether the phoneme is accepted or rejected. Since the HMM fits differently on each phone (the vowels tend to be stable while fricatives are more variable), phone-dependent thresholds are employed.

Experimental Result on the System

We evaluate our model on three different tasks for each of the stages.

For the acoustic model, since we do not have phoneme transcription, we instead evaluate our model on word error rate which indirectly affects the phoneme recognition accuracy. According to our empirical result, our factorized TDNN achieved a WER of 3.81% which relatively 10% outperforms the official report on Kaldi (4.17%) which only applied TDNN.

We evaluate the decoding model based on TIMIT corpus which is a small speech set but with detailed phoneme transcription. We compare the duration of our acoustic-decoding model with the annotation. The mean absolute error is 1.188 frames (36ms on time-domain). Considering the human reaction time is about 300ms, the alignment works well enough for further usage.

We conduct the scoring system evaluation with the CALL_2K speech corpus which contains 2.8 hours of English speech from children English learners. Two professional raters evaluate the pronunciation score of the sentence-level speech in the corpus. To compare our system with the annotation, we average the pronunciation score on phonemes for each sentence and compute the relevance score (i.e. Pearson coefficients, Spearman coefficients, and mutual information coefficients). The results showed in the following table, indicate that our system has been similar to human raters.

The Pronunciation Scoring Experiments

Method-Rater	Pearson-1	Pearson-2	Spearman-1	Spearman-2	MIC-1	MIC-2
Rater 1	\	0.573	\	0.573	\	0.276
GOP	0.425	**0.297**	0.370	**0.315**	0.182	0.143

Shortcomings and Future direction

Based on our discussion, we identify the following shortcomings of our model:

- Domain Mismatch: we now trained our model based on Librispeech which contains only adults. However, our evaluation for the scoring is based on the children's set. Given the truth that children have many specific features to adults, there might be a domain mismatch in the training process.
- Multi-pronunciation: there are some words with multiple accepted pronunciations. However, we cannot handle this in our system now.
- Fixed Sentences: The current system only accepts text&speech pairs for scoring which lack flexibility.
- More Scoring Dimensions: pronunciation is an aspect of scoring, there are also many other scoring directions (e.g. fluency), which are also important for spoken language.
- History Analysis: the current framework only takes sentences individually. as we collect more user activity, we can provide some history analysis (e.g. which phoneme/word they did worst/best)
- No Further Improvement Instructions: though we have the scoring system, we still do not tell the user how they can speak better.

To tackle each of the problems, we think about the following future development directions:

- Domain Mismatch: taking some children's speech datasets in the acoustic model training and considering children speaking features in the scoring model.
- Multi-pronunciation: add multiple paths in the decoding graph to support the different pronunciation or ask the user to state which style they want to learn first.
- Fixed Sentences: add streaming asr to decode and split the speech first and then use the CALL system to score the speech. The method asks us to build a more robust ASR to recognize what users are saying other than current ASR which is adopted as a standard.
- Scoring Dimensions: add more models (e.g. prosody model) to enable more features.
- History Analysis: run a database that stores the history information of the users and create some metrics for a user profile to support history analysis.
- Future Improvement Instructions: deploy a TTS(text-to-speech) module for speech synthesis that can synthesize examples (we also think about using voice conversion to present the speech examples in the user's own voice)