Task 6

To perform ASR for dysarthic speech, I will adopt the methods described in [1] but make some changes to the Audio Event Detection (AED) model. The Xception-based AED model currently has 14 classes. To cater to dysarthic speech, the AED model would be able to categorize 15 classes, with an additional class for 'dysarthic speech' and the original 'speech' class as 'normal speech'. If training from scratch is not possible, a classifier to distinguish between normal speech and dysarthic speech, can be applied to the samples classified as 'speech' by the AED model instead. However, this method is less preferable as dysarthic speech may be misclassified by the original AED model.

The first step of data pre-processing and transformation in the model pipeline would be to convert all the audio data to 16kHz, remove long silences using a Voice Activity Detection (VAD) model, and to segment the audio data into segments of maximum 20 seconds. Then, the data segments are converted into 80-dim log-Mel features and classified using the modified AED model. Modified AED filters and a general filter, as described in [1], can be applied: (i) *Dysarthic-speech-filter*: ignore utterance which no dysarthic speech event; (ii) *Dysarthic-speech-crop*: Crop utterances based on dysarthic speech event to only include the dysarthic speech portion; (iii) *Rand-crop*: random crop for long utterances.

Next, perform self-supervised learning (SSL) pre-training using the Lfb2vec procedure. Results in [1] show that non-streaming SSL pre-training has results similar to supervised streaming pre-training overall, non-streaming SSL will be done. The input is the entire audio segment with the AED filters applied after conversion to log-Mel features. Masking is done by random sampling of initial time steps with probability 0.065, and then masking the subsequent 10 time steps. The masked input is fed to an encoder to yield context vectors. The encoder is trained to re-create the masked portions by optimizing the contrastive loss between masked portions of the context and target vector (from unmasked input), with 100 negative samples drawn randomly from unmasked positions.

As flatNCE with AdamW optimizer is more stable and achieves a lower WER for SSL compared to infoNCE [1], I propose flatNCE as the contrastive loss function with AdamW as the optimizer. Hyperparameters such as max learning rate and steps may depend on the dataset, so experiments can be conducted to choose the best hyperparameters.

After pre-training, perform continuous learning fine-tuning for ASR of dysarthic speech by streaming in new data to update the model weights. Training is done with 20 frames chunk length and 20 frames look ahead length with cross entropy as the loss function on the target language. According to [1], we use a 2-stage fine-tuning approach as the linear projection weights from pre-training cannot initialize the ASR task. In the first stage, freeze the pre-trained encoder weights and only train the linear projection layer from scratch. In the second stage update the weights for the entire model. The decoder uses a language model as described in [1].

(495 words)

References

[1] M. Karimi et al., "Deploying self-supervised learning in the wild for hybrid automatic speech recognition," arXiv.org, https://arxiv.org/abs/2205.08598v1.