Perceptual-Similarity-Aware Deep Speaker Representation Learning for Multi-Speaker Generative Modeling

Yuki Saito, Shinnosuke Takamichi, and Hiroshi Saruwatari (The University of Tokyo, Japan)

ICASSP2022 @ Singapore

©Yuki Saito, May, 2022.

1. Overview

Introduction
- Speaker Embedding (SE): Distributed representation of spkr. ID
- Primal usage: spkr. discriminative tasks e.g., spkr. recognition / verification
- Our focus: spkr. generative tasks e.g., Text-To-Speech (TTS) / Voice Conversion (VC)

Deep speaker representation learning
- DN-based technology for learning SEs
- d-vector and x-vector defined for discriminative tasks

Research question: How can we learn SEs suitable for spkr. generative tasks?

Summary
- Perceptual-similarity-aware SE
  - Trained to represent perceptual similarity among speakers → high interpretability
  - 3 algorithms based on crowdsourced perceptual similarity score matrix
- Human-in-the-loop Active Learning (AL) for high cost-efficiency

2. Methods

Traditional speaker representations

- One-hot speaker code: N_i - dim. discrete vector
  (i.e., ID for pre-stored N_i spkrs.)
  - High similarity when N_i is small
  - Low interpretability & scalability
  - Distance b/w spkrs. ≈ constant
  - Undefined for unseen spkrs.
- d-vector: N_i - dim. continuous vector learned by spkr. recog.
  - Training objective: spkr. recog. loss
  (e.g., softmax cross-entropy or GE2E loss)
  - High scalability
  - Low-dim. SE that can deal with unseen spkrs.
  - Still low interpretability & adaptability in speaker generative tasks

Proposed method: Perceptual-similarity-aware SE learning

- Large scale scoring of perceptual speaker similarity
- Crowdsourcing similarity scores involving 4,000+ listeners
- 1 speaker pair → scored by more than 10 listeners

Instruction of the scoring
- To what degree do these two speakers' voices sound similar?
  (~3: dissimilar - ~1: similar)

- Visualization of obtained similarity scores among 153JP spkrs.
- Matrix representation (speaker similarity score matrix $S$)
- Graph representation (speaker similarity graph $G$)

- Learning SEs based on similarity scores
  - Training embedding DNN to predict perceptual similarity from SEs (i.e., human-oriented machine learning)
  - Algorithm: similarity ($\text{vector, matrix, graph}$) embedding

1. Algorithm 1: similarity vector embedding

2. Algorithm 2: similarity matrix embedding

3. Algorithm 3: similarity graph embedding

3. Experimental evaluations & results

Evaluation 1: interpretability of SEs

- Scatter plots of human-/SE-derived similarity ($r$: PCC)

  - d-vector never considers perceptual similarity.
  - Sim. ($r$) achieves higher PCC than d-vector.
  - Sim. (graph) works the best among the 3 algorithms.

  - AUC of similar speaker-pair detection using similarity b/w SEs

  - AUC of similar speaker-pair detection using similarity b/w SEs

Evaluation 2: quality of multi-speaker generative modeling

- Acoustic model: VAE using pre-trained speech recog. & SE

- Quality of speaker adaptation (AB/XAB tests w/ 25 listeners)
  - A > B: A is significantly better than B ($p < 0.05$).
  - A = B: there is no significant difference b/w A & B.

  - Naturalness

- Controllability of speaker interpolation

  - Horizontal axis: interpolation coefficient
  - Vertical axis: spkr. similarity score

  - Proposed SEs (blue lines) also improve the controllability of d-vector (grey line)

Evaluation 3: cost-efficiency of human-in-the-loop AL

- AL setting: Fully Scored (FS), Partially Scored (PS)
- Curves of similar speaker-pair detection AUC

- Best query strategy: MSF
- AL in Sim. (vec / graph):
  - AUC comparable with FS, while reducing learning/scoring cost.
- AL in Sim. (mat):
  - Degraded AUC in Seen-Unseen spkr.-pair case → due to high data dependency.

- Quality of synthetic speech (DMOS test on spkr. similarity w/ 50 listeners)

  - BOLD scores ≈ FS ($p < 0.05$)
  - DMOs comparable with FS by fewer AL iterations!

- Project page: http://python.org/dem0/JSPS-DCI-1/index.html


(Sources/Materials)