ECE 537 Fundamentals of Speech Processing
Problem Set 10

UNIVERSITY OF ILLINOIS
Department of Electrical and Computer Engineering

Assigned: Sunday, 11/13/2022; Due: Friday, 11/18/2022
Reading: Polyak et al., "Speech Resynthesis from Discrete Disentangled Self-Supervised Representations," 2021

1. In this question, let's analyze the loss gradients of CPC and HuBERT. Assume a simple inner product similarity metric:

$$\text{Score}(c_t, x_t) = x_t^T c_t,$$

where $c_t = [c_{t,1}, \ldots, c_{t,d}]^T$ is the context representation (the output of a transformer), and $x_t = [x_{t,1}, \ldots, x_{t,d}]^T$ is the vector being predicted. Both CPC and HuBERT use a kind of cross-entropy loss,

$$\mathcal{L}_{\text{CE}} = -\sum_t \ln p(x_t|c_t),$$

where $p(x_t|c_t)$ is computed using a softmax:

$$p(x|c_t) = \frac{\exp\left(\text{Score}(c_t, x)\right)}{\sum_{x\in\mathcal{X}_t} \exp\left(\text{Score}(c_t, x)\right)},$$

In both CPC and HuBERT, $x_t \in \mathcal{X}_t$, but CPC and HuBERT differ in the selection of x_t and $\mathcal{X}_t$. In HuBERT, x_t is the codevector to which the MFCC at time t has been quantized, and $\mathcal{X}_t$ is the set of all codevectors. In CPC, x_t is the spectrum (or MFCC, or CNN output) at time t, and $\mathcal{X}_t$ is a set of spectra sampled from different files in the same minibatch.

(a) (1 point) Find the derivative of $\mathcal{L}_{\text{CE}}$ with respect to $c_{t,n}$, the n^{th} element of the Transformer output at time t. Write your answer in terms of $x_{t,n}$, and $\mu_{t,n}$, where $\mu_t = [\mu_{t,1}, \ldots, \mu_{t,d}]^T$ is defined in terms of the softmax outputs $p(x|c_t)$ as

$$\mu_t = \sum_{x\in\mathcal{X}_t} p(x|c_t)x$$

You may or may not find it convenient to use the following form of the gradient of the log softmax:

$$p(i|f) = \frac{\exp(f_i)}{\sum_j \exp(f_j)} \quad \Rightarrow \quad \frac{\partial(-\ln p(i|f))}{\partial f_k} = \begin{cases} p(i|f) - 1 & k = i \\ p(k|f) & k \neq i \end{cases}$$

Solution:

$$\begin{aligned}\frac{\partial \mathcal{L}_{\text{CE}}}{\partial c_{t,n}} &= \sum_{x\in\mathcal{X}_t} \frac{\partial \mathcal{L}_{\text{CE}}}{\partial \text{Score}(c_t,x)} \frac{\partial \text{Score}(c_t,x)}{\partial c_{t,n}} \\ &= \left[(p(x_t|c_t)-1)x_{t,n} + \sum_{x\neq x_t, x\in\mathcal{X}_t} p(x|c_t)x_n\right] \\ &= \mu_{t,n} - x_{t,n}\end{aligned}$$

(b) (1 point) In part (a), you should have discovered that a step in the negative-gradient direction will adjust c_t toward x_t, and away from μ_t. Consider the difference between CPC and HuBERT in the way that μ_t is calculated. How do the differences between CPC and HuBERT affect each step of training? For example, is the gradient lower-dimensional for one than the other? If so, is the subspace chosen randomly, or deterministically?

Solution: For HuBERT, μ_t is a weighted average of the codevectors. If the codevectors are linearly independent, then μ_t is not constrained to any subspace. On the other hand, for CPC, μ_t is a weighted average of the spectra in the negative-sample set. If the number of spectra in the negative-sample set is smaller than the number of dimensions ($|\mathcal{X}_t| < d$), then the gradient is constrained a subspace with $|\mathcal{X}_t|$ dimensions. This subspace is chosen randomly for each minibatch, so over the course of training, the selected subspaces should eventually span the entire space.

2. (1 point) The paper by Polyak et al. resynthesizes speech using HiFi-GAN. In HiFi-GAN, the naturalness of speech is judged by J different discriminators, each of which is a convolutional neural net looking at a different span of speech samples (different dilations, or different durations). Why do you think HiFi-GAN uses many different discriminators, instead of just using one discriminator that takes the entire speech waveform as an input (e.g., using a deep Transformer)?

Solution: I think that what's going on here is that, if the discriminator has the entire speech waveform available in a single view, it will learn to focus on the most obvious differences between natural and synthetic speech—typically short-term correlations, of the kind modeled by LPC. Since those short-term correlations work well most of the time, the discriminator will not learn to pay attention to longer-term correlations. In HiFi-GAN, some of the short-term correlations are not available at the inputs of some of the discriminators (call them the "disadvantaged" discriminators). The disadvantaged discriminators are forced to learn less obvious differences between natural and synthetic speech. Since the discriminators learn to detect those differences, the generator is therefore able to learn how to synthesize speech that better mimics natural speech.