Speech Processing
Exercise 3

Due: 10.6.2019 10:00PM (there will be no extensions!)

1 Guidelines

1. You are not allowed to use external packages other than numpy and scipy.
2. You are allowed to work in pairs.
3. In order to submit your solution please submit the following files:
   (a) `details.txt` - A text file with your full name (in the first line) and ID (in the second line).
   (b) `ex3.py` - The file that contains your main function (attach ANY additional files needed for your code to run).

Follow the instructions and submit all files needed for you code to run.

Good Luck!

Connectionist Temporal Classification

In this exercise you will implement the CTC loss in Python. CTC calculates the probability of a specific labeling given the model's output distribution over phonemes.

Formally, CTC calculates $P(\mathbf{p}|\mathbf{x})$ where $\mathbf{x} = [x_1, x_2, ..., x_T]$ is an input sequence of acoustic features, $\mathbf{p} = [p_1, p_2, ...p_{|\mathbf{p}|}]$ is a sequence of transcription phonemes, and $\mathbf{y}$ is a sequence of network outputs, that is, y^t_k can be interpreted as the probability of observing label k at time t.

Recall, to to calculate the aforementioned probability, we first set

$$\mathbf{z} = [\epsilon, p_1, \epsilon, p_2, \epsilon, ..., p_{|\mathbf{p}|}, \epsilon]$$

Then, we define $\alpha_{s,t}$ to be the probability of the subsequence $\mathbf{z}_{1:s}$ after t time steps. We can calculate α using the following initialization:

$$\alpha_{1,1} = y^1_\epsilon \tag{1}$$

$$\alpha_{2,1} = y^1_{\mathbf{z}_1} \tag{2}$$

$$\alpha_{s,1} = 0, \forall s > 2 \tag{3}$$

and the following dynamic programming:

$$\alpha_{s,t} = \begin{cases} (\alpha_{s-1,t-1} + \alpha_{s,t-1}) \cdot y^t_{\mathbf{z}_s} & \mathbf{z}_s = \epsilon \text{ or } \mathbf{z}_s = \mathbf{z}_{s-2} \\ (\alpha_{s-2,t-1} + \alpha_{s-1,t-1} + \alpha_{s,t-1}) \cdot y^t_{\mathbf{z}_s} & \text{else} \end{cases} \tag{4}$$

Instructions

In this exercise, assume you are given a sequence of phonemes $\mathbf{p}$ and the network's output $\mathbf{y}$. In words, $\mathbf{y}$ is a matrix with the shape of $T \times K$ where T is the number of time steps, and K is the amount of phonemes. Each column i of $\mathbf{y}$ is a distribution over K phoenemes at time i.

Your goal is to implement the CTC function to calculate $P(\mathbf{p}|\mathbf{x})$ using the above equations.

Your code should get 3 arguments:

1. A path to a 2D numpy matrix of network outputs ($\mathbf{y}$). This can be loaded using `numpy.load`.
2. The labeling you wish to calculate the probability for (e.g., "aaabb" means we want the probability of aaabb)
3. A string specifying the possible output tokens (e.g., for an alphabet of [a,b,c] the string should be "abc")

Overall, your code should run with the following command:
"`python ex3.py /some/path/to/mat.npy aaabb abc`"
For your convinience, we attach also an example of inputs (the Submit system will also check against the same inputs/outputs).