EEL 6814
Due Nov 9, 2023

Homework 5

The goal is to utilize a TDNN (with the gamma delay line) and a RNN to distinguish strings of characters belonging to a grammar and compare their performance. A grammar is a rule that establishes the possible sequence of symbols in a string, i.e. the language. In our case the grammar is $\{0^+,1\}$, where $^+$ means one or more zeros. This means that strings like

0,1,0,0,0,1,0,0,1,….

belong to the grammar and strings like

0,0,1,1,0,1,….

do not belong to the grammar. This is a simple case of language processing when the dictionary has only 2 symbols (0 and 1).

In order to train the dynamic neural network, you will create a library of strings that belong to the grammar and counter examples that do NOT belong to the grammar. One of the difficulties is that the number of consecutive zeros is not defined, so there are infinite number of strings belonging to the grammar (and also, not belonging to the grammar). For simplicity let us create distinct strings up to 60 characters' long, with at most 10 consecutive zeros. Also create 60 character strings that do not belong to the grammar by creating strings with more than 2 consecutive 1s (also up to 10 consecutive zeros). This is a little program that you will have to write. You should create up to 500 strings of each class for training. For the TDNN you must select the size of the delay line, and you should adapt the recursive parameter in the context unit). You can experiment with one or two hidden layers, and you can also put memory units in the hidden layers. Use a size of 5 and 20 delays to compare the effect of memory depth in performance. For the RNN use a network with a single recurrent hidden layer. You will need to select the number of units. Must use BPTT as the learning rule.

For testing, I would like you to use the file that is uploaded in the website and quantify the performance of your TDNN and RNN by presenting the results in a confusion matrix. The last column in the file contains the class assignments.