Itay Hassid 209127596 Aviv Shisman 206558157

Ex 4 - report

Requirements

1. Please make sure that data is near the code and divided to: train, valid, test such that in each one you have sub directories with the examples and in test you have another subdirectory called whatever with the tests examples
(for gcommand loader)

2. If you want to save some time you can load our pre trained model we added

Architecture and overview

Our model is similar to deep speech (model we learned in class) architecture
In training function we loop over the data “epoch” times , for each loop we take batches of data and each example in the data we “feed” to the model.

After the example goes through our model , the model outputs a probability over characters for 50 time frames (not 101 like the example), we use ctc loss to maximize the probability of all alignments like learned in class, in validation we perform greedy search to get the predicted word from the probabilities and reduce blanks and repetitives to calculate “CER”.

The model :

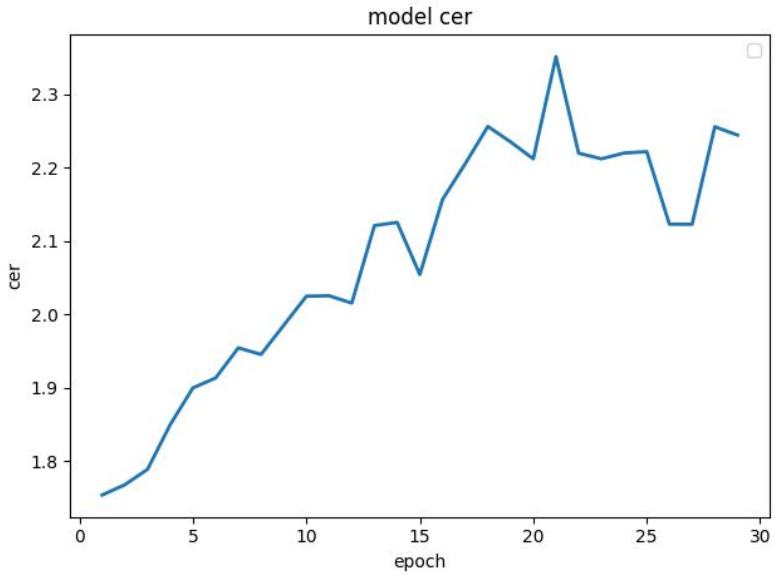
1. Adding noise
2. Perform 1dConv and SELU activation function twice
3. Using avg pool
4. Using 3 layered lstm with dropout
5. Using 2 fully connected layers with SELU and droput
6. Another fully connected and log-softmax

Itay Hassid 209127596 Aviv Shisman 206558157

Evolution of our work:

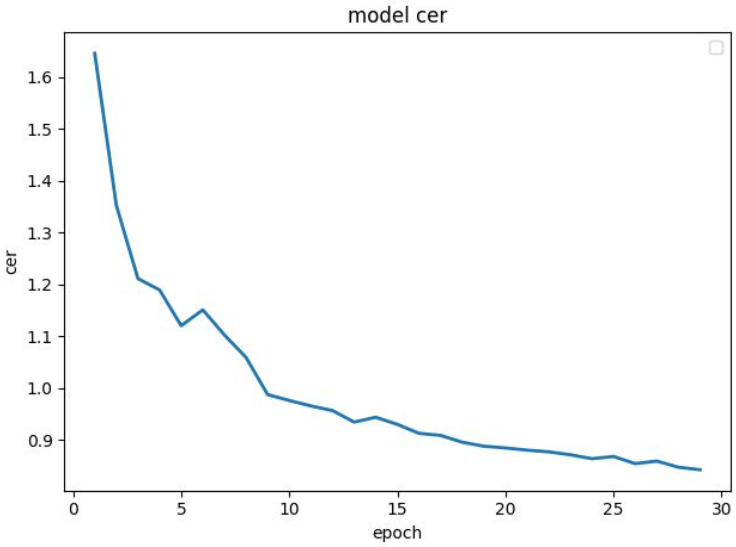
Version 0

Simple dnn and cnn that for each example learn alone its appropriate phone - disaster moving up



Version 1

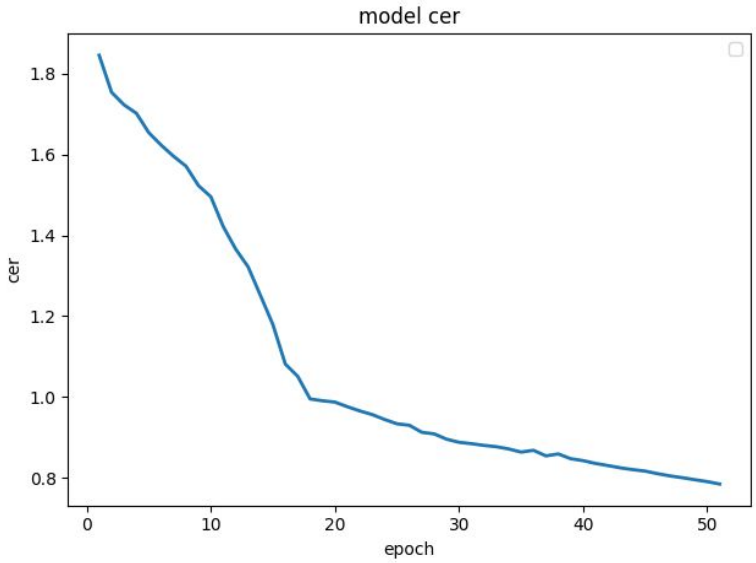
Add rnn, Move to 1d conv, Fix greedy search on output ,and some more bugs , best 0.84



Itay Hassid 209127596 Aviv Shisman 206558157

Version 2

Changed lr (Adam:0.005 to 0.002) and more epochs , best 0.78



Version 3

Selu activation function in dnn

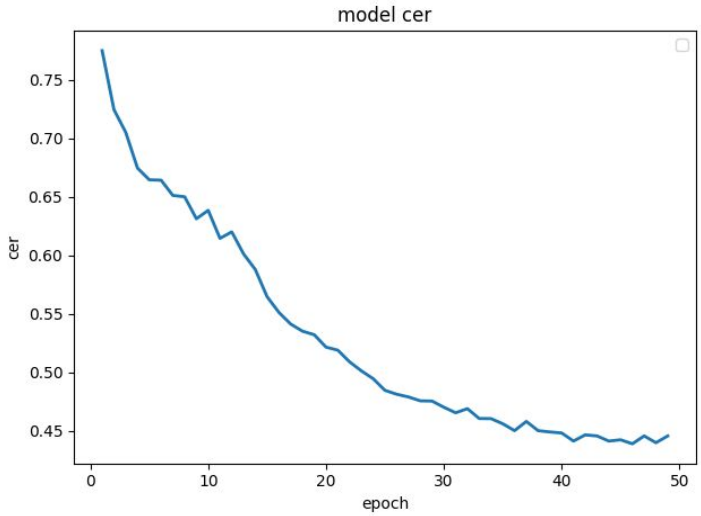
Raise kernel value in conv --7 to 10 and 7

Avg pool doing better than max pool

Lstm - how many times go through the lstm -- 2 to 3

More bug fixes

Best 0.44



Itay Hassid 209127596 Aviv Shisman 206558157

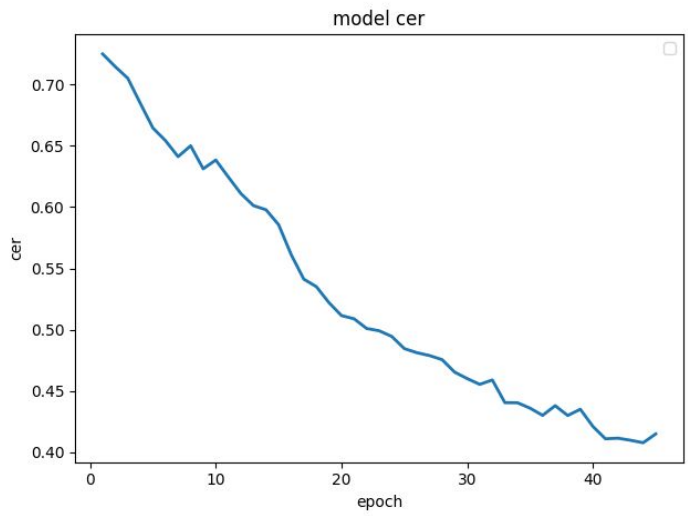
Version 4

Raise number of features in 1d conv

Lstm 3

Dropout in lstm changed to 0.4

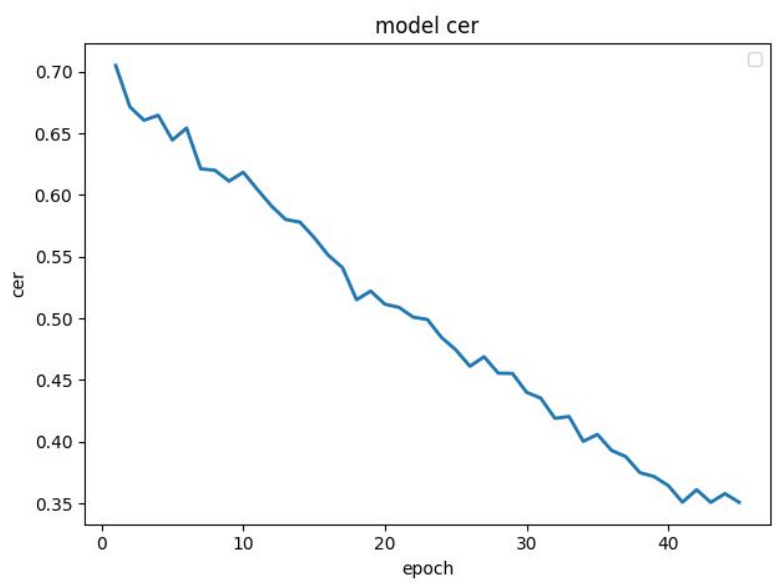
Best 0.4



Version 5

Add noise with normal distribution to the input.

Best 0.35



Itay Hassid 209127596 Aviv Shisman 206558157

Version 6

Change 1d conv to output bigger vectors that eventually get into the lstm model.
(was around 51/52 and now 79 - input size and hidden size) - takes more time

Best 0.31/0.32
(the graph doesn't show it because it required more than 40 epochs and we forgot to save the output in google colab.)