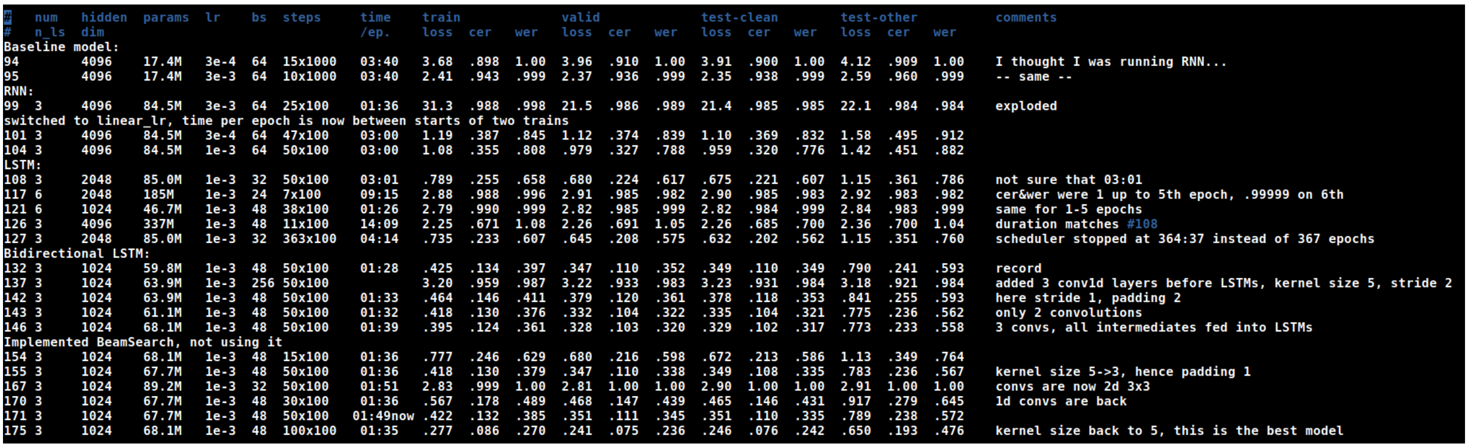
Deep Learning in Audio, Homework 1: Automatic Speech Recognition

Alexey Slizkov
26.10.2023

I have implemented the steps of the instruction, also beam search and beam search with LM (using 3rd party lib). The metrics of BSwLM are significantly better than those without.

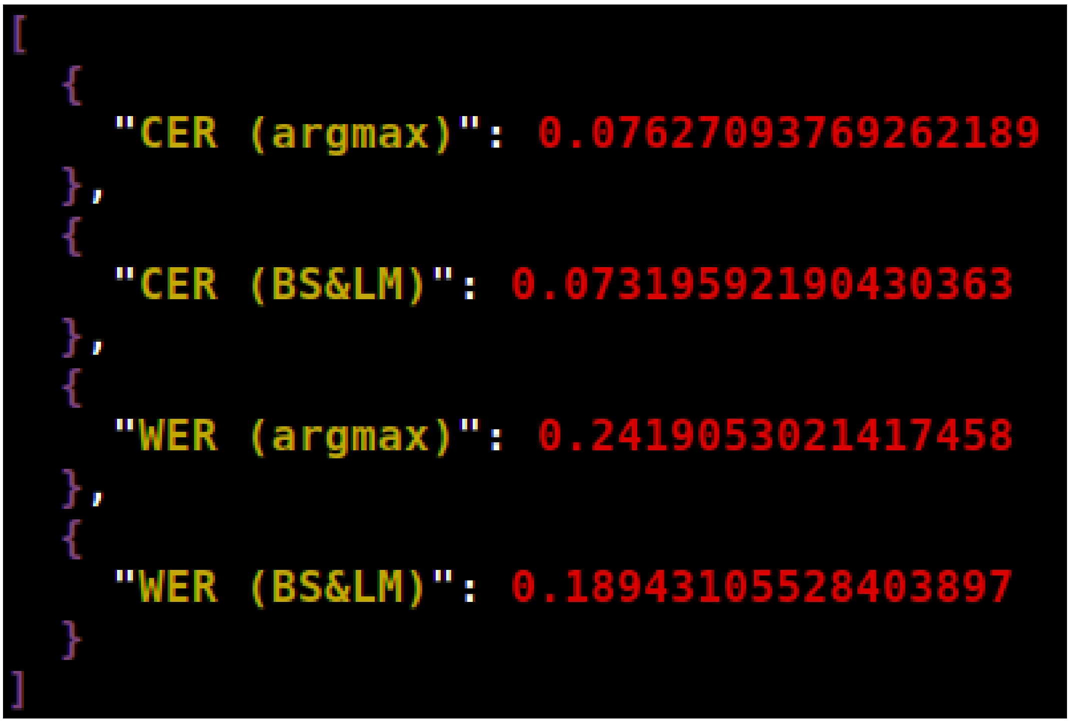
I have a total of 176 wanbd runs, though most of them were interrupted early. Here is the summary of the most successful ones (BS&LM metrics not present here, only plain model):



```
#   num   hidden  params  lr    bs  steps     time    train             valid             test-clean        test-other            comments
#   n_ls  dim                                 /ep.    loss  cer   wer   loss  cer   wer   loss  cer   wer   loss  cer   wer
Baseline model:
94        4096    17.4M   3e-4  64  15x1000   03:40   3.68  .898  1.00  3.96  .910  1.00  3.91  .900  1.00  4.12  .909  1.00    I thought I was running RNN...
95        4096    17.4M   3e-3  64  10x1000   03:40   2.41  .943  .999  2.37  .936  .999  2.35  .938  .999  2.59  .960  .999    -- same --
RNN:
99  3     4096    84.5M   3e-3  64  25x100    01:36   31.3  .988  .998  21.5  .986  .989  21.4  .985  .985  22.1  .984  .984    exploded
switched to linear_lr, time per epoch is now between starts of two trains
101 3     4096    84.5M   3e-4  64  47x100    03:00   1.19  .387  .845  1.12  .374  .839  1.10  .369  .832  1.58  .495  .912
104 3     4096    84.5M   1e-3  64  50x100    03:00   1.08  .355  .808  .979  .327  .788  .959  .320  .776  1.42  .451  .882
LSTM:
108 3     2048    85.0M   1e-3  32  50x100    03:01   .789  .255  .658  .680  .224  .617  .675  .221  .607  1.15  .361  .786    not sure that 03:01
117 6     2048    185M    1e-3  24  7x100     09:15   2.88  .988  .996  2.91  .985  .982  2.90  .985  .983  2.92  .983  .982    cer&wer were 1 up to 5th epoch, .99999 on 6th
121 6     1024    46.7M   1e-3  48  38x100    01:26   2.79  .990  .999  2.82  .985  .999  2.82  .984  .999  2.84  .983  .999    same for 1-5 epochs
126 3     4096    337M    1e-3  48  11x100    14:09   2.25  .671  1.08  2.26  .691  1.05  2.26  .685  .700  2.36  .700  1.04    duration matches #108
127 3     2048    85.0M   1e-3  32  363x100   04:14   .735  .233  .607  .645  .208  .575  .632  .202  .562  1.15  .351  .760    scheduler stopped at 364:37 instead of 367 epochs
Bidirectional LSTM:
132 3     1024    59.8M   1e-3  48  50x100    01:28   .425  .134  .397  .347  .110  .352  .349  .110  .349  .790  .241  .593    record
137 3     1024    63.9M   1e-3  256 50x100            3.20  .959  .987  3.22  .933  .983  3.23  .931  .984  3.18  .921  .984    added 3 conv1d layers before LSTMs, kernel size 5, stride 2
142 3     1024    63.9M   1e-3  48  50x100    01:33   .464  .146  .411  .379  .120  .361  .378  .118  .353  .841  .255  .593    here stride 1, padding 2
143 3     1024    61.1M   1e-3  48  50x100    01:32   .418  .130  .376  .332  .104  .322  .335  .104  .321  .775  .236  .562    only 2 convolutions
146 3     1024    68.1M   1e-3  48  50x100    01:39   .395  .124  .361  .328  .103  .320  .329  .102  .317  .773  .233  .558    3 convs, all intermediates fed into LSTMs
Implemented BeamSearch, not using it
154 3     1024    68.1M   1e-3  48  15x100    01:36   .777  .246  .629  .680  .216  .598  .672  .213  .586  1.13  .349  .764
155 3     1024    67.7M   1e-3  48  50x100    01:36   .418  .130  .379  .347  .110  .338  .349  .108  .335  .783  .236  .567    kernel size 5->3, hence padding 1
167 3     1024    89.2M   1e-3  32  50x100    01:51   2.83  .999  1.00  2.81  1.00  1.00  2.90  1.00  1.00  2.91  1.00  1.00    convs are now 2d 3x3
170 3     1024    67.7M   1e-3  48  30x100    01:36   .567  .178  .489  .468  .147  .439  .465  .146  .431  .917  .279  .645    1d convs are back
171 3     1024    67.7M   1e-3  48  50x100   01:49now .422  .132  .385  .351  .111  .345  .351  .110  .335  .789  .238  .572
175 3     1024    68.1M   1e-3  48  100x100   01:35   .277  .086  .270  .241  .075  .236  .246  .076  .242  .650  .193  .476    kernel size back to 5, this is the best model
```

I claim a 0.189 test-clean WER, please check output.json, the output of test.py.

More precisely, I have:



```
[
  {
    "CER (argmax)": 0.07627093769262189
  },
  {
    "CER (BS&LM)": 0.07319592190430363
  },
  {
    "WER (argmax)": 0.2419053021417458
  },
  {
    "WER (BS&LM)": 0.18943105528403897
  }
]
```

Please check the graphs and all the statistics of my runs in wandb, the relevant ones are listen at the first picture.

If you want to reproduce and run train.py and test.py yourself, please check the readme file.