**MODELS**

What to try:

* ATTNN paper + Regul
* ATT NN paper but with dsconv e gdsconv
* Riassunto cap 9.3.5
* 40 MFCC + Depthwise separable CNN FIG4 from Hello Edge: Keyword Spotting on Microcontrollers
  + DSconv = DepthwiseConv2D --> batch norm --> Relu --> 1x1Conv2D --> batch norm --> Relu
* 40 MFCC: DEEP RESIDUAL LEARNING FOR SMALL-FOOTPRINT KEYWORD SPOTTING
* TC-ResNet8 from Temporal Convolution for Real-time Keyword Spotting on Mobile Devices
* If particular erors: (no-on) train a different network based on those errors
* NO transformer for audio classification (<https://codeburst.io/how-to-use-transformer-for-audio-classification-5f4bc0d0c1f0>, <https://mc.ai/how-to-use-the-transformer-for-audio-classification%E2%80%8A-%E2%80%8Apart-2/>, <https://towardsdatascience.com/music-genre-classification-transformers-vs-recurrent-neural-networks-631751a71c58> )

|  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **#** | **Dataset** | **Acc** | **Bias** | **Var** | **Params** | **Network** | **LR** | **OPT** | **Reg** | **batchS** | **Comments** | **File** |
| 1 | 10 cmd  10k-1k-1k  40mfcc +delta | 0.771 | 0 | 0.227 | 1,465,226 | LeNet5 con elu | / | Nadam | / | 32 | 0 Bias  High Variance  Overfitting |  |
| 2 | 10 cmd  10k-1k-1k  40mfcc +delta | 0.687 | 0.013 | 0.333 | 123,542 | LeNet5 con elu  Senza i 2 FC layers | / | Nadam | / | 32 | 0 Bias  High Variance  Overfitting |  |
| 3 | 10 cmd  10k-1k-1k  40mfcc +delta | 0.656 | 0 | 0.358 | 123,630 | LeNet5 con elu  Senza i 2 FC layers  Con Batch Norm | / | Nadam | / | 32 | Peggio del 2 | 2020-07-19\_15-13\_LeNet5-elu-1FC-BN |
| 4 | 10 cmd  10k-1k-1k  40mfcc +delta | 0.691 | 0.002 | 0.312 | 123,542 | LeNet5 con elu  Senza i 2 FC layers  Regolariz | / | Nadam | L2  1e-4 | 32 | Leggermente meglio del 2  Overfitting | 2020-07-19\_15-36\_LeNet5-elu-1FC-Reg |
| 5 | 10 cmd  10k-1k-1k  40mfcc +delta | 0.707 | 0.081 | 0.235 | 123,542 | LeNet5 con elu  Senza i 2 FC layers  Regolariz | / | Nadam | L2  1e-3 | 32 | Some Bias  High Variance  Leggermente meglio del 2 e 3  Overfitting | 2020-07-19\_15-44\_LeNet5-elu-1FC-Reg |
| 6 | 10 cmd  10k-1k-1k  40mfcc +delta | 0.796 | 0 | 0.223 | 1,455,422 | LeNet5 con elu  Con FC 120  Regolariz | / | Nadam | L2  1e-3 | 32 | 0 Bias  High Variance  Meglio dei precedenti  Overfitting | 2020-07-19\_16-02\_LeNet5-elu-2FC-Reg |
| 7 | 10 cmd  10k-1k-1k  40mfcc +delta | 0.796 | 0 | 0.223 | 1,455,422 | LeNet5 con elu  Con FC 120  Regolariz | / | Nadam | L2  1e-2 | 32 | Cambiato nulla dal 6 | 2020-07-19\_16-44\_LeNet5-elu-2FC-Reg |
| 8 | 10 cmd  10k-1k-1k  12mfcc +delta | 0.757 | 0 | 0.254 | 224,341 | ATTNETWORK paper | / | Nadam | / | 32 | 0 Bias  High Variance  Overfitting | 2020-07-19\_18-07\_AttRNNSpeechModel |
| 9 | 10 cmd  10k-1k-1k  80 mel | 0.755 | 0 | 0.251 | 224,341 | ATTNETWORK paper | decay | Nadam | / | 32 | 0 Bias  High Variance  Overfitting  Confusion Matrix seems slightly better Than 8 | 2020-07-20\_14-58\_AttRNNSpeechModel |
| 10 | 10 cmd  10k-1k-1k  80 mel Normalized |  |  |  | 224,341 | ATTNETWORK paper | decay | Nadam | / | 32 | Vary Bad |  |
| 11 | 10 cmd  10k-1k-1k  80 mel | 0.802 | 0 | 0.223 | 224,341 | ATTNETWORK paper | decay | adam | / | 32 | 0 Bias  High Variance  Overfitting  Better than 6 | 2020-07-20\_15-34\_AttRNNSpeechModel |
| 12 | 10 cmd  10k-1k-1k  40mfcc | 0.773 | 0 | 0.232 | 224,341 | ATTNETWORK paper | decay | adam | / | 32 | 0 Bias  High Variance  Overfitting | 2020-07-20\_16-40\_AttRNNSpeechModel |
| 13 | 10 cmd  20k-1k-1k  40mfcc +delta | 0.773 | 0.029 | 0.205 | 224,341 | ATTNETWORK paper | / | adam | / | 32 | Little Bias  High Variance  Overfitting  Little variance improvement | CLUSTER DEI  2020-07-20\_22-28\_AttRNNSpeechModel |
| 14 | 10 cmd  30k-3k-3k  80 mel | 0.874 | 0.008 | 0.128 | 224,341 | ATTNETWORK paper | / | Nadam | / | 32 | 0 Bias  Little variance | CLUSTER DEI  2020-07-21\_17-04\_AttRNNSpeechModel |
| 15 | 10 cmd  30k-3k-3k  80 mel | 0.909 | 0 | 0.099 | 224,341 | ATTNETWORK paper | decay | adam | / | 32 | 0 Bias  Little variance | CLUSTER DEI  2020-07-21\_19-28\_AttRNNSpeechModel |
| 16 | 10 cmd  30k-3k-3k  40mfcc +delta | ??? | runn | ??? | 224,341 | ATTNETWORK paper | decay | adam | / | 32 | ??? RUNNING | CLUSTER DEI |
| 17 | 10 cmd  1k-1k-1k  No preprocess | 0.643 | 0 | 0.363 | 256,058 | 1D CNN Paper  relu | / | adam | / | 32 | 0 Bias  High Variance  Overfitting  Solution: mor train data. 18 | 2020-07-22\_16-48\_DirectCNN |
| 18 | 10 cmd  10k-1k-1k  No preprocess | 0.858 | 0.006 | 0.149 | 256,058 | 1D CNN Paper  relu | / | adam | / | 32 | 0 Bias  Little variance  Solution: mor train data. 19 | 2020-07-22\_17-00\_DirectCNN |
| 19 | 10 cmd  20k-2k-2k  No preprocess | 0.888 | 0.018 | 0.1 | 256,058 | 1D CNN Paper  relu | / | adam | / | 32 | 0 Bias  Little variance  Solution: mor train data. 20 | 2020-07-22\_17-15\_DirectCNN |
| 20 | 10 cmd  30k-3k-3k  No preprocess | 0.912 | 0.015 | 0.086 | 256,058 | 1D CNN Paper  relu | / | adam | / | 32 | 0 Bias  Little variance  True – Predic: (off-up), (up-off), (go-down), (right-left), (yes-left), (no-go) | 2020-07-22\_17-26\_DirectCNN |
| 21 | 10 cmd  30k-3k-3k  No preprocess | 0.917 | 0.013 | 0.081 | 256,058 | 1D CNN Paper  relu | / | Nadam | / | 32 | 0 Bias  Little variance  True – Predic: (off-up), (up-off), | 2020-07-22\_17-51\_DirectCNN |
| 22 | 10 cmd  30k-3k-3k  No preprocess | 0.914 | 0.01 | 0.089 | 256,058 | 1D CNN Paper  relu | / | Nadam | / | 100 | 0 Bias  Little variance  True – Predic: (off-up), (up-off), (go-no), (no-go) | 2020-07-22\_18-14\_DirectCNN |
| 23 | 10 cmd  30k-3k-3k  No preprocess | 0.93 | 0.017 | 0.058 | 257,018 | 1D CNN Paper  Relu  BatchNorm after each CNN  Dropout 0.25 after 128 e 64 | / | Nadam | / | 32 | 0 Bias  Little variance  Solution: regularization. 24  True – Predic: (off-up), (up-off), (go-no), (no-go) | 2020-07-23\_13-11\_DirectCNNBatchDrop |
| 24 | 10 cmd  30k-3k-3k  No preprocess | 0.875 | 0.147 | 0.015 | 257,018 | 1D CNN Paper  Relu  BatchNorm after each CNN  Dropout 0.25 after 128 e 64 | / | Nadam | L2  1e-2 | 32 | High bias  0 Variance  Very bad plot for validation  Solution: reduce regularization. 25 | 2020-07-23\_14-08\_DirectCNNBatchDrop |
| 25 | 10 cmd  30k-3k-3k  No preprocess | 0.926 | 0.079 | 0.088 | 257,018 | 1D CNN Paper  Relu  BatchNorm after each CNN  Dropout 0.25 after 128 e 64 | / | Nadam | L2  1e-3 | 32 | Little Bias  Little Variance  Very bad plot for validation  Solution: reduce regularization. 26 | 2020-07-23\_14-41\_DirectCNNBatchDropRegu |
| 26 | 10 cmd  30k-3k-3k  No preprocess | 0.925 | 0.046 | 0.053 | 257,018 | 1D CNN Paper  Relu  BatchNorm after each CNN  Dropout 0.25 after 128 e 64 | / | Nadam | L2  1e-4 | 32 | Little Bias  Little Variance  Very bad plot for validation  Solution: reduce regularization. 27 | 2020-07-23\_15-30\_DirectCNNBatchDropRegu |
| 27 | 10 cmd  30k-3k-3k  No preprocess | 0.919 | 0.03 | 0.061 | 257,018 | 1D CNN Paper  Relu  BatchNorm after each CNN  Dropout 0.25 after 128 e 64 | / | Nadam | L2  1e-5 | 32 | Little Bias  Little Variance  Meglio il 23 senza regolarizzazione | 2020-07-23\_15-56\_DirectCNNBatchDropRegu |
| 28 | 10 cmd  30k-3k-3k  No preprocess | 0.929 | 0.013 | 0.112 | 257,018 | 1D CNN Paper  elu  BatchNorm after each CNN  Dropout 0.25 after 128 e 64 | / | Nadam | / | 32 |  | 2020-07-23\_16-33\_DirectCNNBatchDropELU |
|  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |

**Datasets**:

* 10 cmd
* 10 cmd + silence +unknown
* 20 cmd + unknown

**Preprocessing**:

* No. Directly train on the 16000-element vector
* Mel spectrogram with 80mels
* 40 MFCC
* 40MFCC +delta +deltadelta
* Solo 13 MFCC?????

**Architectures:**

* ????

**Metrics**:

* Accuracy (in a problem where there is a large class imbalance, a model can predict the value of the majority class for all predictions and achieve a high classification accuracy. So, further performance measures are needed such as F1 score and Brier score, but since in our dataset the classes are balanced we can still use Accuracy)
* Prediction speed (ms)

Comparison:

* DEEP RESIDUAL LEARNING FOR SMALL-FOOTPRINT KEYWORD SPOTTING

Project (60 points):

* originality (10)
* data preprocessing techniques (10)
* learning architectures (20)
* comparison against other/existing approaches (10)
* live demo of the code (10)

Written report (40 points):

* clarity of exposition (10)
* completeness (10)
* analysis of results (number and type of metrics used) (20)

Oral exposition (20 points):

* duration (your talk must be shorter than 25 minutes, using slides) (10)
* clarity of exposition (10)

The final grade will be computed as grade = (points\*30)/100

the proposed architecture shown in Figure 2 is made of four convolutional layers, possibly interlaced with max pooling layers, followed by two fully connected layers and an output layer.

input an array of 16,000 dimensions, which represents 1-second of audio sampled at 16kHz.

The proposed 1D CNN has large receptive fields in the first convolutional layers since it is assumed that the first layers should have a more global view of the audio signal.

The output of the last pooling layer for all feature maps is flattened and used as input to a fully connected layer.

In order to reduce the over-fitting, batch normalization is applied after the activation function of each convolution layer.

after the last pooling layer, there are two fully connected layers with 128 and 64 neurons respectively on which a drop-out is applied with a probability of 0.25 for both layers.

By the use of the architecture shown in Figure 2, it is possible to omit a signal processing module because the network is powerful enough to extract relevant low-level and high-level information from the audio waveform.