Use rectified signals. (rectify\_emg\_moving\_average(X,20))

Use cost sensitive learning(1:5) for binary classification(0:others)

Drop some files out for test. No shuffle and split the rest data as 80% for training and 20% for validation.

Use residual block in conv1d structure.

## Drop Files[6,30,31,32,33,34,35]:

Class 0 : others

Train (acc 0.987)

|  |  |  |
| --- | --- | --- |
|  | Predicted 0 | Predicted others |
| Actual 0 | 5781 | 77 |
| Actual others | 2 | 560 |

Valid (acc 0.946)

|  |  |  |
| --- | --- | --- |
|  | Predicted 0 | Predicted others |
| Actual 0 | 1466 | 78 |
| Actual others | 13 | 140 |

Test (acc 0.889)

|  |  |  |
| --- | --- | --- |
|  | Predicted 0 | Predicted others |
| Actual 0 | 273 | 53 |
| Actual others | 5 | 196 |

Class 1 : 2 : 6

Train (acc 0.816)

|  |  |  |  |
| --- | --- | --- | --- |
|  | Predicted 1 | Predicted 2 | Predicted 6 |
| Actual 1 | 74 | 14 | 32 |
| Actual 2 | 37 | 232 | 9 |
| Actual 6 | 10 | 1 | 153 |

Valid (acc 0.736)

|  |  |  |  |
| --- | --- | --- | --- |
|  | Predicted 1 | Predicted 2 | Predicted 6 |
| Actual 1 | 13 | 10 | 10 |
| Actual 2 | 8 | 68 | 1 |
| Actual 6 | 8 | 3 | 31 |

Test (acc 0.800)

|  |  |  |  |
| --- | --- | --- | --- |
|  | Predicted 1 | Predicted 2 | Predicted 6 |
| Actual 1 | 43 | 4 | 5 |
| Actual 2 | 18 | 38 | 1 |
| Actual 6 | 12 | 0 | 80 |

Class 2 : 6

Train (acc 0.950)

|  |  |  |
| --- | --- | --- |
|  | Predicted 2 | Predicted 6 |
| Actual 2 | 261 | 17 |
| Actual 6 | 5 | 159 |

Valid (acc 0.932)

|  |  |  |
| --- | --- | --- |
|  | Predicted 2 | Predicted 6 |
| Actual 2 | 72 | 5 |
| Actual 6 | 3 | 39 |

Test (acc 0.919)

|  |  |  |
| --- | --- | --- |
|  | Predicted 2 | Predicted 6 |
| Actual 2 | 47 | 10 |
| Actual 6 | 2 | 90 |

Class 1 : 6

Train (acc 0.826)

|  |  |  |
| --- | --- | --- |
|  | Predicted 1 | Predicted 6 |
| Actual 1 | 82 | 38 |
| Actual 6 | 4 | 160 |

Valid (acc 0.773)

|  |  |  |
| --- | --- | --- |
|  | Predicted 1 | Predicted 6 |
| Actual 1 | 20 | 13 |
| Actual 6 | 4 | 38 |

Test (acc 0.826)

|  |  |  |
| --- | --- | --- |
|  | Predicted 1 | Predicted 6 |
| Actual 1 | 39 | 13 |
| Actual 6 | 12 | 80 |

Class 1 : 2

Train (acc 0.876)

|  |  |  |
| --- | --- | --- |
|  | Predicted 1 | Predicted 2 |
| Actual 1 | 108 | 12 |
| Actual 2 | 37 | 241 |

Valid (acc 0.872)

|  |  |  |
| --- | --- | --- |
|  | Predicted 1 | Predicted 2 |
| Actual 1 | 24 | 9 |
| Actual 2 | 5 | 72 |

Test (acc 0.706)

|  |  |  |
| --- | --- | --- |
|  | Predicted 1 | Predicted 2 |
| Actual 1 | 47 | 5 |
| Actual 2 | 27 | 30 |

## Drop Files[7,30,31,32,33,34,35]:

Class 0 : others

Train (acc 0.983)

|  |  |  |
| --- | --- | --- |
|  | Predicted 0 | Predicted others |
| Actual 0 | 5796 | 102 |
| Actual others | 0 | 467 |

Valid (acc 0.945)

|  |  |  |
| --- | --- | --- |
|  | Predicted 0 | Predicted others |
| Actual 0 | 1472 | 82 |
| Actual others | 9 | 120 |

Test (acc 0.971)

|  |  |  |
| --- | --- | --- |
|  | Predicted 0 | Predicted others |
| Actual 0 | 267 | 9 |
| Actual others | 8 | 312 |

Class 1 : 2 : 6

Train (acc 0.850)

|  |  |  |  |
| --- | --- | --- | --- |
|  | Predicted 1 | Predicted 2 | Predicted 6 |
| Actual 1 | 18 | 20 | 23 |
| Actual 2 | 15 | 234 | 6 |
| Actual 6 | 4 | 2 | 145 |

Valid (acc 0.759)

|  |  |  |  |
| --- | --- | --- | --- |
|  | Predicted 1 | Predicted 2 | Predicted 6 |
| Actual 1 | 5 | 7 | 6 |
| Actual 2 | 7 | 62 | 3 |
| Actual 6 | 5 | 3 | 31 |

Test (acc 0.557)

|  |  |  |  |
| --- | --- | --- | --- |
|  | Predicted 1 | Predicted 2 | Predicted 6 |
| Actual 1 | 21 | 30 | 75 |
| Actual 2 | 20 | 49 | 16 |
| Actual 6 | 0 | 0 | 108 |

Class 2 : 6

Train (acc 0.965)

|  |  |  |
| --- | --- | --- |
|  | Predicted 2 | Predicted 6 |
| Actual 2 | 243 | 12 |
| Actual 6 | 2 | 149 |

Valid (acc 0.909)

|  |  |  |
| --- | --- | --- |
|  | Predicted 2 | Predicted 6 |
| Actual 2 | 67 | 5 |
| Actual 6 | 5 | 34 |

Test (acc 0.886)

|  |  |  |
| --- | --- | --- |
|  | Predicted 2 | Predicted 6 |
| Actual 2 | 63 | 22 |
| Actual 6 | 0 | 108 |

Class 1 : 6

Train (acc 0.867)

|  |  |  |
| --- | --- | --- |
|  | Predicted 1 | Predicted 6 |
| Actual 1 | 38 | 23 |
| Actual 6 | 5 | 146 |

Valid (acc 0.719)

|  |  |  |
| --- | --- | --- |
|  | Predicted 1 | Predicted 6 |
| Actual 1 | 9 | 9 |
| Actual 6 | 7 | 32 |

Test (acc 0.692)

|  |  |  |
| --- | --- | --- |
|  | Predicted 1 | Predicted 6 |
| Actual 1 | 54 | 72 |
| Actual 6 | 0 | 108 |

Class 1 : 2

Train (acc 0.927)

|  |  |  |
| --- | --- | --- |
|  | Predicted 1 | Predicted 2 |
| Actual 1 | 49 | 12 |
| Actual 2 | 11 | 244 |

Valid (acc 0.911)

|  |  |  |
| --- | --- | --- |
|  | Predicted 1 | Predicted 2 |
| Actual 1 | 15 | 3 |
| Actual 2 | 5 | 67 |

Test (acc 0.649)

|  |  |  |
| --- | --- | --- |
|  | Predicted 1 | Predicted 2 |
| Actual 1 | 76 | 50 |
| Actual 2 | 24 | 61 |

## Drop Files[5,30,31,32,33,34,35]:

Class 0 : others

Train (acc 0.974)

|  |  |  |
| --- | --- | --- |
|  | Predicted 0 | Predicted others |
| Actual 0 | 5705 | 165 |
| Actual others | 0 | 629 |

Valid (acc 0.929)

|  |  |  |
| --- | --- | --- |
|  | Predicted 0 | Predicted others |
| Actual 0 | 1431 | 116 |
| Actual others | 5 | 166 |

Test (acc 0.939)

|  |  |  |
| --- | --- | --- |
|  | Predicted 0 | Predicted others |
| Actual 0 | 291 | 20 |
| Actual others | 6 | 110 |

Class 1 : 2 : 6

Train (acc 0.817)

|  |  |  |  |
| --- | --- | --- | --- |
|  | Predicted 1 | Predicted 2 | Predicted 6 |
| Actual 1 | 83 | 44 | 24 |
| Actual 2 | 27 | 285 | 4 |
| Actual 6 | 16 | 0 | 146 |

Valid (acc 0.817)

|  |  |  |  |
| --- | --- | --- | --- |
|  | Predicted 1 | Predicted 2 | Predicted 6 |
| Actual 1 | 23 | 11 | 7 |
| Actual 2 | 9 | 78 | 0 |
| Actual 6 | 3 | 1 | 38 |

Test (acc 0.362)

|  |  |  |  |
| --- | --- | --- | --- |
|  | Predicted 1 | Predicted 2 | Predicted 6 |
| Actual 1 | 1 | 12 | 0 |
| Actual 2 | 2 | 5 | 2 |
| Actual 6 | 42 | 16 | 36 |

Class 2 : 6

Train (acc 0.974)

|  |  |  |
| --- | --- | --- |
|  | Predicted 2 | Predicted 6 |
| Actual 2 | 310 | 6 |
| Actual 6 | 6 | 156 |

Valid (acc 0.968)

|  |  |  |
| --- | --- | --- |
|  | Predicted 2 | Predicted 6 |
| Actual 2 | 85 | 2 |
| Actual 6 | 2 | 40 |

Test (acc 0.708)

|  |  |  |
| --- | --- | --- |
|  | Predicted 2 | Predicted 6 |
| Actual 2 | 6 | 3 |
| Actual 6 | 27 | 67 |

Class 1 : 6

Train (acc 0.833)

|  |  |  |
| --- | --- | --- |
|  | Predicted 1 | Predicted 6 |
| Actual 1 | 125 | 26 |
| Actual 6 | 26 | 136 |

Valid (acc 0.891)

|  |  |  |
| --- | --- | --- |
|  | Predicted 1 | Predicted 6 |
| Actual 1 | 34 | 7 |
| Actual 6 | 2 | 40 |

Test (acc 0.429)

|  |  |  |
| --- | --- | --- |
|  | Predicted 1 | Predicted 6 |
| Actual 1 | 13 | 0 |
| Actual 6 | 61 | 33 |

Class 1 : 2

Train (acc 0.937)

|  |  |  |
| --- | --- | --- |
|  | Predicted 1 | Predicted 2 |
| Actual 1 | 144 | 7 |
| Actual 2 | 22 | 294 |

Valid (acc 0.875)

|  |  |  |
| --- | --- | --- |
|  | Predicted 1 | Predicted 2 |
| Actual 1 | 30 | 11 |
| Actual 2 | 5 | 82 |

Test (acc 0.318)

|  |  |  |
| --- | --- | --- |
|  | Predicted 1 | Predicted 2 |
| Actual 1 | 1 | 12 |
| Actual 2 | 3 | 6 |

## Model:

def residual\_block(x, i):

tanh\_out = layers.Conv1D(filters,

2,

dilation\_rate = 2\*\*i,

padding='causal',

name='dilated\_conv\_%d\_tanh' % (2 \*\* i),

activation='tanh'

)(x)

sigm\_out = layers.Conv1D(filters,

2,

dilation\_rate = 2\*\*i,

padding='causal',

name='dilated\_conv\_%d\_sigm' % (2 \*\* i),

activation='sigmoid'

)(x)

z = layers.Multiply(name='gated\_activation\_%d' % (i))([tanh\_out, sigm\_out])

skip = layers.Conv1D(filters, 1, name='skip\_%d'%(i))(z)

res = layers.Add(name='residual\_block\_%d' % (i))([skip, x])

return res, skip

x = layers.Input(shape=(1024,8), name='original\_input')

skip\_connections = []

out = layers.Conv1D(filters, 2, dilation\_rate=1, padding='causal', name='dilated\_conv\_1')(x)

for i in range(1, 6):

out, skip = residual\_block(out,i)

skip\_connections.append(skip)

out = layers.Add(name='skip\_connections')(skip\_connections)

out = layers.Activation('elu')(out)

out = layers.Conv1D(32, 3, strides = 1, padding='same', activation = 'relu')(out)

out = layers.MaxPooling1D(2, padding='same')(out)

out = layers.Conv1D(16, 7, padding='same', activation='elu')(out)

out = layers.MaxPooling1D(2, padding='same)(out)

out = layers.Conv1D(8, 3, activation='elu', padding='same')(out)

out = layers.GlobalAveragePooling1D()(out)

out = layers.Dense(2,activation='softmax')(out)

model = Model(x, out)