In [1]:
```
import numpy as np
import tensorflow as tf
import tensorflow.keras.backend as K
from mpl_toolkits.mplot3d import Axes3D
import matplotlib.pyplot as plt
import pandas as pd
import os
import datetime
import argparse
from VAE_functions import *
from NILM_functions import *
import pickle
from scipy.stats import norm
from keras.utils.vis_utils import plot_model
from dtw import *
import logging
import json
```

```
2022-12-22 23:54:51.428742: I tensorflow/core/platform/cpu_feature_guard.
cc:193] This TensorFlow binary is optimized with oneAPI Deep Neural Netwo
rk Library (oneDNN) to use the following CPU instructions in performance-
critical operations:  SSE4.1 SSE4.2
To enable them in other operations, rebuild TensorFlow with the appropria
te compiler flags.
```

In [6]:
```
```

In [12]:
```
ADD_VAL_SET = False

logging.getLogger('tensorflow').disabled = True

################################################################################
# Config
################################################################################
#parser = argparse.ArgumentParser()

#parser.add_argument("--gpu", default=0, type=int, help="Appliance to lea
#parser.add_argument("--config", default="", type=str, help="Path to the
#a = parser.parse_args()
a=argparse.Namespace(config='Config/House_2/WashingMachine_VAE.json', gpu

# Select GPU
os.environ["CUDA_DEVICE_ORDER"] = "PCI_BUS_ID"
os.environ["CUDA_VISIBLE_DEVICES"] = str(a.gpu)

print("################################################################")
print("NILM DISAGREGATOR")
print("GPU : {}".format(a.gpu))
print("CONFIG : {}".format(a.config))
print("################################################################")
```

```
################################################################################
######
NILM DISAGREGATOR
GPU : 0
CONFIG : Config/House_2/WashingMachine_VAE.json
################################################################################
######
```

In [15]:
```
with open(a.config) as data_file:
    nilm = json.load(data_file)

np.random.seed(123)

name = "NILM_Disag_{}".format(nilm["appliance"])
time = datetime.datetime.now().strftime("%Y%m%d-%H%M%S")

for r in range(1, nilm["run"]+1):
    ######################################################################
    # Load dataset
    ######################################################################
    x_train, y_train = load_data(nilm["model"], nilm["appliance"], nilm["

    main_mean = nilm["preprocessing"]["main_mean"]
    main_std = nilm["preprocessing"]["main_std"]

    app_mean = nilm["preprocessing"]["app_mean"]
    app_std = nilm["preprocessing"]["app_std"]

    ######################################################################
    # Training parameters
    ######################################################################
    epochs = nilm["training"]["epoch"]
    batch_size = nilm["training"]["batch_size"]

    STEPS_PER_EPOCH = x_train.shape[0]//batch_size

    lr_schedule = tf.keras.optimizers.schedules.InverseTimeDecay(
                    float(nilm["training"]["lr"]),
                    decay_steps=STEPS_PER_EPOCH*nilm["training"]["decay_s
                    decay_rate=1,
                    staircase=False)

    ######################################################################
    # Optimizer
    ######################################################################
    def get_optimizer(opt):
        if opt == "adam":
            return tf.keras.optimizers.Adam(lr_schedule)
        else:
            return tf.keras.optimizers.RMSprop(lr_schedule)


    ######################################################################
    # Create and initialize the model
    ######################################################################
    if nilm["model"] == "VAE":
        model = create_model(nilm["model"], nilm["config"], nilm["preproc
    elif nilm["model"] == "DAE":
        model = create_model(nilm["model"], nilm["config"], nilm["preproc
    elif nilm["model"] == "S2P":
        model = create_model(nilm["model"], nilm["config"], nilm["preproc
    elif nilm["model"] == "S2S":
        model = create_model(nilm["model"], nilm["config"], nilm["preproc

    ######################################################################
    # Callback checkpoint settings
    ######################################################################
```

```
list_callbacks = []

# Create a callback that saves the model's weights
if nilm["training"]["save_best"] == 1:
    checkpoint_path = "{}/{}/{}/logs/model/House_{}/{}/{}".format(nam
    checkpoint_dir = os.path.dirname(checkpoint_path)

    cp_callback = tf.keras.callbacks.ModelCheckpoint(filepath=checkpo
                                                     save_weights_onl
                                                     verbose=0,
                                                     monitor="val_mea
                                                     mode="min",
                                                     save_best_only=T
else:
    checkpoint_path = "{}/{}/{}/logs/model/House_{}/{}/{}".format(nam
    checkpoint_dir = os.path.dirname(checkpoint_path)

    cp_callback = tf.keras.callbacks.ModelCheckpoint(filepath=checkpo
                                                     save_weights_onl
                                                     verbose=0,
                                                     period=1)

list_callbacks.append(cp_callback)

if nilm["training"]["patience"] > 0:
    patience = nilm["training"]["patience"]
    start_epoch = nilm["training"]["start_stopping"]

    print("Patience : {}, Start at : {}".format(patience, start_epoch

    es_callback = CustomStopper(monitor='val_loss', patience=patience

    list_callbacks.append(es_callback)

#########################################################################
# Normalize Test Data and History Callback
#########################################################################
if ADD_VAL_SET:
    if nilm["dataset"]["name"] == "ukdale":
        if nilm["model"] == "S2P":
            x_test_s2p, y_test_s2p = transform_s2p(x_test, y_test, ni
            history_cb = AdditionalValidationSets([((x_test_s2p-main_
        else:
            history_cb = AdditionalValidationSets([((x_test-main_mean

    elif nilm["dataset"]["name"] == "house_2":
        history_cb = AdditionalValidationSets([(x_test, y_test, 'Hous
    elif nilm["dataset"]["name"] == "refit":
        history_cb = AdditionalValidationSets([(x_test, y_test, 'Hous

    list_callbacks.append(history_cb)

#########################################################################
# Summary of all parameters
#########################################################################
print("#################################################################
print("Summary")
print("#################################################################
print("{}".format(nilm))
print("Run number : {}/{}".format(r,nilm["run"]))
```

```
    print("############################################################

    if not os.path.exists("{}/{}/{}/logs/model/House_{}/{}".format(name,
        os.makedirs("{}/{}/{}/logs/model/House_{}/{}".format(name, nilm["

    with open("{}/{}/{}/logs/model/House_{}/{}/config.txt".format(name, n
        json.dump(nilm, outfile)

    ############################################################################
    # Train Model
    ############################################################################
    if nilm["dataset"]["name"] == "ukdale":
        ########################################################################
        # Real Validation
        ########################################################################
        if nilm["model"] == "S2P":
            x_train_s2p, y_train_s2p = transform_s2p(x_train, y_train, ni

            history = model.fit((x_train_s2p-main_mean)/main_std, (y_trai
                                 epochs=epochs, batch_size=batch_size, cal

        elif nilm["model"] == "VAE":
            history = model.fit((x_train-main_mean)/main_std, (y_train-ap
                                 epochs=epochs, batch_size=batch_size, cal

        elif nilm["model"] == "S2S":
            history = model.fit((x_train-main_mean)/main_std, (y_train-ap
                                 epochs=epochs, batch_size=batch_size, cal

        elif nilm["model"] == "DAE":
            history = model.fit((x_train-main_mean)/main_std, (y_train-ap
                                 epochs=epochs, batch_size=batch_size, cal

        ########################################################################
        # Save history
        ########################################################################
        np.save("{}/{}/{}/logs/model/House_{}/{}/{}/history.npy".format(n
        #np.save("{}/{}/{}/logs/model/{}/{}/history_cb_{}.npy".format(nam

        print("Fit finished!")
    else:
        print("Error in dataset name!")
```

```
############################################################################
######
Create train dataset
Total house 1 : x:(35910, 1024, 1), y:(35910, 1024, 1)
Ratio house 1 : 0.15, x:(5386, 1024, 1), y:(5386, 1024, 1)
Total house 5 : x:(7260, 1024, 1), y:(7260, 1024, 1)
Ratio house 5 : 1, x:(7260, 1024, 1), y:(7260, 1024, 1)
Complete dataset : x:(12646, 1024, 1), y(12646, 1024, 1)
2022-12-23 00:06:13.714831: I tensorflow/core/platform/cpu_feature_guard.
cc:193] This TensorFlow binary is optimized with oneAPI Deep Neural Netwo
rk Library (oneDNN) to use the following CPU instructions in performance-
critical operations:  SSE4.1 SSE4.2
To enable them in other operations, rebuild TensorFlow with the appropria
te compiler flags.
```

```
Model: "model"
_________________________________________________________________
 Layer (type)                Output Shape              Param #   Connected to
==================================================================================================
 input_1 (InputLayer)        [(None, 1024, 1)]         0         []

 conv_seq1_Conv1D1 (Conv1D)  (None, 1024, 64)          256       ['input_1[0][0]']

 conv_seq1_BatchNorm1 (BatchNor  (None, 1024, 64)      256       ['conv_seq1_Conv1D1[0][0]']
 malization)

 conv_seq1_ReLU1 (Activation)  (None, 1024, 64)        0         ['conv_seq1_BatchNorm1[0][0]']

 conv_seq1_Conv1D2 (Conv1D)  (None, 1024, 64)          4160      ['conv_seq1_ReLU1[0][0]']

 conv_seq1_BatchNorm2 (BatchNor  (None, 1024, 64)      256       ['conv_seq1_Conv1D2[0][0]']
 malization)

 conv_seq1_ReLU2 (Activation)  (None, 1024, 64)        0         ['conv_seq1_BatchNorm2[0][0]']

 conv_seq1_Conv1D3 (Conv1D)  (None, 1024, 256)         49408     ['conv_seq1_ReLU2[0][0]']

 conv_seq1_BatchNorm3 (BatchNor  (None, 1024, 256)     1024      ['conv_seq1_Conv1D3[0][0]']
 malization)

 conv_seq1_InstNorm2 (InstanceN  (None, 1024, 256)     2         ['conv_seq1_BatchNorm3[0][0]']
 ormalization)

 conv_seq1_ReLU3 (Activation)  (None, 1024, 256)       0         ['conv_seq1_InstNorm2[0][0]']

 pool1 (MaxPooling1D)        (None, 512, 256)          0         ['conv_seq1_ReLU3[0][0]']

 conv_seq2_Conv1D1 (Conv1D)  (None, 512, 64)           49216     ['pool1[0][0]']

 conv_seq2_BatchNorm1 (BatchNor  (None, 512, 64)       256       ['conv_seq2_Conv1D1[0][0]']
 malization)

 conv_seq2_ReLU1 (Activation)  (None, 512, 64)         0         ['conv_seq2_BatchNorm1[0][0]']

 conv_seq2_Conv1D2 (Conv1D)  (None, 512, 64)           4160      ['conv_seq2_ReLU1[0][0]']

 conv_seq2_BatchNorm2 (BatchNor  (None, 512, 64)       256       ['conv_seq2
```

```
eq2_Conv1D2[0][0]']
 malization)

 conv_seq2_ReLU2 (Activation)   (None, 512, 64)      0           ['conv_s
eq2_BatchNorm2[0][0]']

 conv_seq2_Conv1D3 (Conv1D)     (None, 512, 256)     49408       ['conv_s
eq2_ReLU2[0][0]']

 conv_seq2_BatchNorm3 (BatchNor  (None, 512, 256)    1024        ['conv_s
eq2_Conv1D3[0][0]']
 malization)

 add (Add)                      (None, 512, 256)     0           ['conv_s
eq2_BatchNorm3[0][0]',
                                                                 'pool1
[0][0]']

 conv_seq2_InstNorm2 (InstanceN  (None, 512, 256)    2           ['add
[0][0]']
 ormalization)

 conv_seq2_ReLU3 (Activation)   (None, 512, 256)     0           ['conv_s
eq2_InstNorm2[0][0]']

 pool2 (MaxPooling1D)           (None, 256, 256)     0           ['conv_s
eq2_ReLU3[0][0]']

 conv_seq3_Conv1D1 (Conv1D)     (None, 256, 64)      49216       ['pool2
[0][0]']

 conv_seq3_BatchNorm1 (BatchNor  (None, 256, 64)     256         ['conv_s
eq3_Conv1D1[0][0]']
 malization)

 conv_seq3_ReLU1 (Activation)   (None, 256, 64)      0           ['conv_s
eq3_BatchNorm1[0][0]']

 conv_seq3_Conv1D2 (Conv1D)     (None, 256, 64)      4160        ['conv_s
eq3_ReLU1[0][0]']

 conv_seq3_BatchNorm2 (BatchNor  (None, 256, 64)     256         ['conv_s
eq3_Conv1D2[0][0]']
 malization)

 conv_seq3_ReLU2 (Activation)   (None, 256, 64)      0           ['conv_s
eq3_BatchNorm2[0][0]']

 conv_seq3_Conv1D3 (Conv1D)     (None, 256, 256)     49408       ['conv_s
eq3_ReLU2[0][0]']

 conv_seq3_BatchNorm3 (BatchNor  (None, 256, 256)    1024        ['conv_s
eq3_Conv1D3[0][0]']
 malization)

 add_1 (Add)                    (None, 256, 256)     0           ['conv_s
eq3_BatchNorm3[0][0]',
                                                                 'pool2
[0][0]']
```

```
 conv_seq3_InstNorm2 (InstanceN  (None, 256, 256)    2           ['add_1
[0][0]']
 ormalization)

 conv_seq3_ReLU3 (Activation)   (None, 256, 256)     0           ['conv_s
eq3_InstNorm2[0][0]']

 pool3 (MaxPooling1D)           (None, 128, 256)     0           ['conv_s
eq3_ReLU3[0][0]']

 conv_seq4_Conv1D1 (Conv1D)     (None, 128, 64)      49216       ['pool3
[0][0]']

 conv_seq4_BatchNorm1 (BatchNor  (None, 128, 64)     256         ['conv_s
eq4_Conv1D1[0][0]']
 malization)

 conv_seq4_ReLU1 (Activation)   (None, 128, 64)      0           ['conv_s
eq4_BatchNorm1[0][0]']

 conv_seq4_Conv1D2 (Conv1D)     (None, 128, 64)      4160        ['conv_s
eq4_ReLU1[0][0]']

 conv_seq4_BatchNorm2 (BatchNor  (None, 128, 64)     256         ['conv_s
eq4_Conv1D2[0][0]']
 malization)

 conv_seq4_ReLU2 (Activation)   (None, 128, 64)      0           ['conv_s
eq4_BatchNorm2[0][0]']

 conv_seq4_Conv1D3 (Conv1D)     (None, 128, 256)     49408       ['conv_s
eq4_ReLU2[0][0]']

 conv_seq4_BatchNorm3 (BatchNor  (None, 128, 256)    1024        ['conv_s
eq4_Conv1D3[0][0]']
 malization)

 add_2 (Add)                    (None, 128, 256)     0           ['conv_s
eq4_BatchNorm3[0][0]',
                                                                 'pool3
[0][0]']

 conv_seq4_InstNorm2 (InstanceN  (None, 128, 256)    2           ['add_2
[0][0]']
 ormalization)

 conv_seq4_ReLU3 (Activation)   (None, 128, 256)     0           ['conv_s
eq4_InstNorm2[0][0]']

 pool4 (MaxPooling1D)           (None, 64, 256)      0           ['conv_s
eq4_ReLU3[0][0]']

 conv_seq5_Conv1D1 (Conv1D)     (None, 64, 64)       49216       ['pool4
[0][0]']

 conv_seq5_BatchNorm1 (BatchNor  (None, 64, 64)      256         ['conv_s
eq5_Conv1D1[0][0]']
 malization)

 conv_seq5_ReLU1 (Activation)   (None, 64, 64)       0           ['conv_s
```

```
eq5_BatchNorm1[0][0]']

 conv_seq5_Conv1D2 (Conv1D)     (None, 64, 64)       4160        ['conv_s
eq5_ReLU1[0][0]']

 conv_seq5_BatchNorm2 (BatchNor  (None, 64, 64)      256         ['conv_s
eq5_Conv1D2[0][0]']
 malization)

 conv_seq5_ReLU2 (Activation)   (None, 64, 64)       0           ['conv_s
eq5_BatchNorm2[0][0]']

 conv_seq5_Conv1D3 (Conv1D)     (None, 64, 256)      49408       ['conv_s
eq5_ReLU2[0][0]']

 conv_seq5_BatchNorm3 (BatchNor  (None, 64, 256)     1024        ['conv_s
eq5_Conv1D3[0][0]']
 malization)

 add_3 (Add)                    (None, 64, 256)      0           ['conv_s
eq5_BatchNorm3[0][0]',
                                                                  'pool4
[0][0]']

 conv_seq5_InstNorm2 (InstanceN  (None, 64, 256)     2           ['add_3
[0][0]']
 ormalization)

 conv_seq5_ReLU3 (Activation)   (None, 64, 256)      0           ['conv_s
eq5_InstNorm2[0][0]']

 pool5 (MaxPooling1D)           (None, 32, 256)      0           ['conv_s
eq5_ReLU3[0][0]']

 conv_seq6_Conv1D1 (Conv1D)     (None, 32, 64)       49216       ['pool5
[0][0]']

 conv_seq6_BatchNorm1 (BatchNor  (None, 32, 64)      256         ['conv_s
eq6_Conv1D1[0][0]']
 malization)

 conv_seq6_ReLU1 (Activation)   (None, 32, 64)       0           ['conv_s
eq6_BatchNorm1[0][0]']

 conv_seq6_Conv1D2 (Conv1D)     (None, 32, 64)       4160        ['conv_s
eq6_ReLU1[0][0]']

 conv_seq6_BatchNorm2 (BatchNor  (None, 32, 64)      256         ['conv_s
eq6_Conv1D2[0][0]']
 malization)

 conv_seq6_ReLU2 (Activation)   (None, 32, 64)       0           ['conv_s
eq6_BatchNorm2[0][0]']

 conv_seq6_Conv1D3 (Conv1D)     (None, 32, 256)      49408       ['conv_s
eq6_ReLU2[0][0]']

 conv_seq6_BatchNorm3 (BatchNor  (None, 32, 256)     1024        ['conv_s
eq6_Conv1D3[0][0]']
 malization)
```

```
 add_4 (Add)                    (None, 32, 256)      0           ['conv_s
eq6_BatchNorm3[0][0]',
                                                                 'pool5
[0][0]']

 conv_seq6_ReLU3 (Activation)   (None, 32, 256)      0           ['add_4
[0][0]']

 pool6 (MaxPooling1D)           (None, 16, 256)      0           ['conv_s
eq6_ReLU3[0][0]']

 conv_seq7_Conv1D1 (Conv1D)     (None, 16, 64)       49216       ['pool6
[0][0]']

 conv_seq7_BatchNorm1 (BatchNor  (None, 16, 64)      256         ['conv_s
eq7_Conv1D1[0][0]']
 malization)

 conv_seq7_ReLU1 (Activation)   (None, 16, 64)       0           ['conv_s
eq7_BatchNorm1[0][0]']

 conv_seq7_Conv1D2 (Conv1D)     (None, 16, 64)       4160        ['conv_s
eq7_ReLU1[0][0]']

 conv_seq7_BatchNorm2 (BatchNor  (None, 16, 64)      256         ['conv_s
eq7_Conv1D2[0][0]']
 malization)

 conv_seq7_ReLU2 (Activation)   (None, 16, 64)       0           ['conv_s
eq7_BatchNorm2[0][0]']

 conv_seq7_Conv1D3 (Conv1D)     (None, 16, 256)      49408       ['conv_s
eq7_ReLU2[0][0]']

 conv_seq7_BatchNorm3 (BatchNor  (None, 16, 256)     1024        ['conv_s
eq7_Conv1D3[0][0]']
 malization)

 add_5 (Add)                    (None, 16, 256)      0           ['conv_s
eq7_BatchNorm3[0][0]',
                                                                 'pool6
[0][0]']

 conv_seq7_ReLU3 (Activation)   (None, 16, 256)      0           ['add_5
[0][0]']

 flatten (Flatten)              (None, 4096)         0           ['conv_s
eq7_ReLU3[0][0]']

 z_log_var (Dense)              (None, 16)           65552       ['flatte
n[0][0]']

 z_sigma (Lambda)               (None, 16)           0           ['z_log_
var[0][0]']

 eps (InputLayer)               [(None, 16)]         0           []

 z_mu (Dense)                   (None, 16)           65552       ['flatte
n[0][0]']
```

```
 z_eps (Multiply)               (None, 16)           0           ['z_sigm
a[0][0]',
                                                                 'eps
[0][0]']

 z (Add)                        (None, 16)           0           ['z_mu
[0][0]',
                                                                 'z_eps
[0][0]']

 reshape1 (Reshape)             (None, 16, 1)        0           ['z
[0][0]']

 dconv_seq4_Conv1D1 (Conv1D)    (None, 16, 64)       256         ['reshap
e1[0][0]']

 dconv_seq4_BatchNorm1 (BatchNo  (None, 16, 64)      256         ['dconv_
seq4_Conv1D1[0][0]']
 rmalization)

 dconv_seq4_ReLU1 (Activation)  (None, 16, 64)       0           ['dconv_
seq4_BatchNorm1[0][0]']

 dconv_seq4_Conv1D2 (Conv1D)    (None, 16, 64)       4160        ['dconv_
seq4_ReLU1[0][0]']

 dconv_seq4_BatchNorm2 (BatchNo  (None, 16, 64)      256         ['dconv_
seq4_Conv1D2[0][0]']
 rmalization)

 dconv_seq4_ReLU2 (Activation)  (None, 16, 64)       0           ['dconv_
seq4_BatchNorm2[0][0]']

 dconv_seq4_Conv1D3 (Conv1D)    (None, 16, 256)      49408       ['dconv_
seq4_ReLU2[0][0]']

 dconv_seq4_BatchNorm3 (BatchNo  (None, 16, 256)     1024        ['dconv_
seq4_Conv1D3[0][0]']
 rmalization)

 dconv_seq4_ReLU3 (Activation)  (None, 16, 256)      0           ['dconv_
seq4_BatchNorm3[0][0]']

 dconc5 (Concatenate)           (None, 16, 512)      0           ['dconv_
seq4_ReLU3[0][0]',
                                                                 'conv_s
eq7_ReLU3[0][0]']

 lambda (Lambda)                (None, 16, 1, 512)   0           ['dconc5
[0][0]']

 conv2d_transpose (Conv2DTransp  (None, 32, 1, 256)  393472      ['lambda
[0][0]']
 ose)

 lambda_1 (Lambda)              (None, 32, 256)      0           ['conv2d
_transpose[0][0]']

 dconv_seq5_Conv1D1 (Conv1D)    (None, 32, 64)       49216       ['lambda
```

```
_1[0][0]']

 dconv_seq5_BatchNorm1 (BatchNo  (None, 32, 64)      256         ['dconv_
seq5_Conv1D1[0][0]']
 rmalization)

 dconv_seq5_ReLU1 (Activation)  (None, 32, 64)       0           ['dconv_
seq5_BatchNorm1[0][0]']

 dconv_seq5_Conv1D2 (Conv1D)    (None, 32, 64)       4160        ['dconv_
seq5_ReLU1[0][0]']

 dconv_seq5_BatchNorm2 (BatchNo  (None, 32, 64)      256         ['dconv_
seq5_Conv1D2[0][0]']
 rmalization)

 dconv_seq5_ReLU2 (Activation)  (None, 32, 64)       0           ['dconv_
seq5_BatchNorm2[0][0]']

 dconv_seq5_Conv1D3 (Conv1D)    (None, 32, 256)      49408       ['dconv_
seq5_ReLU2[0][0]']

 dconv_seq5_BatchNorm3 (BatchNo  (None, 32, 256)     1024        ['dconv_
seq5_Conv1D3[0][0]']
 rmalization)

 add_6 (Add)                    (None, 32, 256)      0           ['dconv_
seq5_BatchNorm3[0][0]',
                                                                 'lambda
_1[0][0]']

 dconv_seq5_ReLU3 (Activation)  (None, 32, 256)      0           ['add_6
[0][0]']

 dconc7 (Concatenate)           (None, 32, 512)      0           ['dconv_
seq5_ReLU3[0][0]',
                                                                 'conv_s
eq6_ReLU3[0][0]']

 lambda_2 (Lambda)              (None, 32, 1, 512)   0           ['dconc7
[0][0]']

 conv2d_transpose_1 (Conv2DTran  (None, 64, 1, 256)  393472      ['lambda
_2[0][0]']
 spose)

 lambda_3 (Lambda)              (None, 64, 256)      0           ['conv2d
_transpose_1[0][0]']

 dconv_seq6_Conv1D1 (Conv1D)    (None, 64, 64)       49216       ['lambda
_3[0][0]']

 dconv_seq6_BatchNorm1 (BatchNo  (None, 64, 64)      256         ['dconv_
seq6_Conv1D1[0][0]']
 rmalization)

 dconv_seq6_ReLU1 (Activation)  (None, 64, 64)       0           ['dconv_
seq6_BatchNorm1[0][0]']

 dconv_seq6_Conv1D2 (Conv1D)    (None, 64, 64)       4160        ['dconv_
```

```
seq6_ReLU1[0][0]']

 dconv_seq6_BatchNorm2 (BatchNo  (None, 64, 64)      256         ['dconv_
seq6_Conv1D2[0][0]']
 rmalization)

 dconv_seq6_ReLU2 (Activation)  (None, 64, 64)       0           ['dconv_
seq6_BatchNorm2[0][0]']

 dconv_seq6_Conv1D3 (Conv1D)    (None, 64, 256)      49408       ['dconv_
seq6_ReLU2[0][0]']

 dconv_seq6_BatchNorm3 (BatchNo  (None, 64, 256)     1024        ['dconv_
seq6_Conv1D3[0][0]']
 rmalization)

 add_7 (Add)                    (None, 64, 256)      0           ['dconv_
seq6_BatchNorm3[0][0]',
                                                                 'lambda
_3[0][0]']

 dconv_seq6_ReLU3 (Activation)  (None, 64, 256)      0           ['add_7
[0][0]']

 dconc9 (Concatenate)           (None, 64, 512)      0           ['dconv_
seq6_ReLU3[0][0]',
                                                                 'conv_s
eq5_ReLU3[0][0]']

 lambda_4 (Lambda)              (None, 64, 1, 512)   0           ['dconc9
[0][0]']

 conv2d_transpose_2 (Conv2DTran  (None, 128, 1, 256)  393472     ['lambda
_4[0][0]']
 spose)

 lambda_5 (Lambda)              (None, 128, 256)     0           ['conv2d
_transpose_2[0][0]']

 dconv_seq7_Conv1D1 (Conv1D)    (None, 128, 64)      49216       ['lambda
_5[0][0]']

 dconv_seq7_BatchNorm1 (BatchNo  (None, 128, 64)     256         ['dconv_
seq7_Conv1D1[0][0]']
 rmalization)

 dconv_seq7_ReLU1 (Activation)  (None, 128, 64)      0           ['dconv_
seq7_BatchNorm1[0][0]']

 dconv_seq7_Conv1D2 (Conv1D)    (None, 128, 64)      4160        ['dconv_
seq7_ReLU1[0][0]']

 dconv_seq7_BatchNorm2 (BatchNo  (None, 128, 64)     256         ['dconv_
seq7_Conv1D2[0][0]']
 rmalization)

 dconv_seq7_ReLU2 (Activation)  (None, 128, 64)      0           ['dconv_
seq7_BatchNorm2[0][0]']

 dconv_seq7_Conv1D3 (Conv1D)    (None, 128, 256)     49408       ['dconv_
```

```
seq7_ReLU2[0][0]']

 dconv_seq7_BatchNorm3 (BatchNo  (None, 128, 256)    1024        ['dconv_
seq7_Conv1D3[0][0]']
 rmalization)

 add_8 (Add)                    (None, 128, 256)     0           ['dconv_
seq7_BatchNorm3[0][0]',
                                                                 'lambda
_5[0][0]']

 dconv_seq7_ReLU3 (Activation)  (None, 128, 256)     0           ['add_8
[0][0]']

 dconc11 (Concatenate)          (None, 128, 512)     0           ['dconv_
seq7_ReLU3[0][0]',
                                                                 'conv_s
eq4_ReLU3[0][0]']

 lambda_6 (Lambda)              (None, 128, 1, 512)  0           ['dconc1
1[0][0]']

 conv2d_transpose_3 (Conv2DTran  (None, 256, 1, 256)  393472     ['lambda
_6[0][0]']
 spose)

 lambda_7 (Lambda)              (None, 256, 256)     0           ['conv2d
_transpose_3[0][0]']

 dconv_seq8_Conv1D1 (Conv1D)    (None, 256, 64)      49216       ['lambda
_7[0][0]']

 dconv_seq8_BatchNorm1 (BatchNo  (None, 256, 64)     256         ['dconv_
seq8_Conv1D1[0][0]']
 rmalization)

 dconv_seq8_ReLU1 (Activation)  (None, 256, 64)      0           ['dconv_
seq8_BatchNorm1[0][0]']

 dconv_seq8_Conv1D2 (Conv1D)    (None, 256, 64)      4160        ['dconv_
seq8_ReLU1[0][0]']

 dconv_seq8_BatchNorm2 (BatchNo  (None, 256, 64)     256         ['dconv_
seq8_Conv1D2[0][0]']
 rmalization)

 dconv_seq8_ReLU2 (Activation)  (None, 256, 64)      0           ['dconv_
seq8_BatchNorm2[0][0]']

 dconv_seq8_Conv1D3 (Conv1D)    (None, 256, 256)     49408       ['dconv_
seq8_ReLU2[0][0]']

 dconv_seq8_BatchNorm3 (BatchNo  (None, 256, 256)    1024        ['dconv_
seq8_Conv1D3[0][0]']
 rmalization)

 add_9 (Add)                    (None, 256, 256)     0           ['dconv_
seq8_BatchNorm3[0][0]',
                                                                 'lambda
_7[0][0]']
```

```
 dconv_seq8_ReLU3 (Activation)  (None, 256, 256)     0           ['add_9
[0][0]']

 dconc13 (Concatenate)          (None, 256, 512)     0           ['dconv_
seq8_ReLU3[0][0]',
                                                                 'conv_s
eq3_ReLU3[0][0]']

 lambda_8 (Lambda)              (None, 256, 1, 512)  0           ['dconc1
3[0][0]']

 conv2d_transpose_4 (Conv2DTran  (None, 512, 1, 256)  393472     ['lambda
_8[0][0]']
 spose)

 lambda_9 (Lambda)              (None, 512, 256)     0           ['conv2d
_transpose_4[0][0]']

 dconv_seq9_Conv1D1 (Conv1D)    (None, 512, 64)      49216       ['lambda
_9[0][0]']

 dconv_seq9_BatchNorm1 (BatchNo  (None, 512, 64)     256         ['dconv_
seq9_Conv1D1[0][0]']
 rmalization)

 dconv_seq9_ReLU1 (Activation)  (None, 512, 64)      0           ['dconv_
seq9_BatchNorm1[0][0]']

 dconv_seq9_Conv1D2 (Conv1D)    (None, 512, 64)      4160        ['dconv_
seq9_ReLU1[0][0]']

 dconv_seq9_BatchNorm2 (BatchNo  (None, 512, 64)     256         ['dconv_
seq9_Conv1D2[0][0]']
 rmalization)

 dconv_seq9_ReLU2 (Activation)  (None, 512, 64)      0           ['dconv_
seq9_BatchNorm2[0][0]']

 dconv_seq9_Conv1D3 (Conv1D)    (None, 512, 256)     49408       ['dconv_
seq9_ReLU2[0][0]']

 dconv_seq9_BatchNorm3 (BatchNo  (None, 512, 256)    1024        ['dconv_
seq9_Conv1D3[0][0]']
 rmalization)

 add_10 (Add)                   (None, 512, 256)     0           ['dconv_
seq9_BatchNorm3[0][0]',
                                                                 'lambda
_9[0][0]']

 dconv_seq9_ReLU3 (Activation)  (None, 512, 256)     0           ['add_10
[0][0]']

 dconc15 (Concatenate)          (None, 512, 512)     0           ['dconv_
seq9_ReLU3[0][0]',
                                                                 'conv_s
eq2_ReLU3[0][0]']

 lambda_10 (Lambda)             (None, 512, 1, 512)  0           ['dconc1
```

```
5[0][0']

 conv2d_transpose_5 (Conv2DTran  (None, 1024, 1, 256  393472     ['lambda
_10[0][0']
 spose)                         )

 lambda_11 (Lambda)             (None, 1024, 256)    0           ['conv2d
_transpose_5[0][0']

 dconv_seq10_Conv1D1 (Conv1D)   (None, 1024, 64)     49216       ['lambda
_11[0][0']

 dconv_seq10_BatchNorm1 (BatchN  (None, 1024, 64)    256         ['dconv_
seq10_Conv1D1[0][0']
 ormalization)

 dconv_seq10_ReLU1 (Activation)  (None, 1024, 64)    0           ['dconv_
seq10_BatchNorm1[0][0']

 dconv_seq10_Conv1D2 (Conv1D)   (None, 1024, 64)     4160        ['dconv_
seq10_ReLU1[0][0']

 dconv_seq10_BatchNorm2 (BatchN  (None, 1024, 64)    256         ['dconv_
seq10_Conv1D2[0][0']
 ormalization)

 dconv_seq10_ReLU2 (Activation)  (None, 1024, 64)    0           ['dconv_
seq10_BatchNorm2[0][0']

 dconv_seq10_Conv1D3 (Conv1D)   (None, 1024, 256)    49408       ['dconv_
seq10_ReLU2[0][0']

 dconv_seq10_BatchNorm3 (BatchN  (None, 1024, 256)   1024        ['dconv_
seq10_Conv1D3[0][0']
 ormalization)

 add_11 (Add)                   (None, 1024, 256)    0           ['dconv_
seq10_BatchNorm3[0][0',
                                                                 'lambda
_11[0][0']

 dconv_seq10_ReLU3 (Activation)  (None, 1024, 256)   0           ['add_11
[0][0']

 dconc17 (Concatenate)          (None, 1024, 512)    0           ['dconv_
seq10_ReLU3[0][0',
                                                                 'conv_s
eq1_ReLU3[0][0']

 x_pred (Conv1D)                (None, 1024, 1)      1537        ['dconc1
7[0][0']

==================================================================================
Total params: 3,856,043
Trainable params: 3,845,291
Non-trainable params: 10,752
__________________________________________________________________________________
Patience : 10, Start at : 5
```

```
##########################################################################
######
Summary
##########################################################################
######
{'model': 'VAE', 'config': 0, 'appliance': 'WashingMachine', 'run': 10, '
training': {'batch_size': 32, 'epoch': 100, 'lr': 0.001, 'decay_steps':
2, 'optimizer': 'rmsprop', 'patience': 10, 'start_stopping': 5, 'save_bes
t': 0, 'ratio_train': 0.3, 'ratio_test': 0, 'validation_split': 0.2, 'S2P
_strides': 1}, 'dataset': {'name': 'ukdale', 'test': {'house': [2], 'rati
o': [1]}, 'train': {'house': [1, 5], 'ratio': [0.15, 1]}}, 'preprocessing
': {'main_mean': 0, 'main_std': 1, 'app_mean': 0, 'app_std': 1, 'width':
1024, 'strides': 256}}
Run number : 1/10
##########################################################################
######
Epoch 1/100
```

```
---------------------------------------------------------------------------
ValueError                                Traceback (most recent call las
/tmp/ipykernel_117241/761872173.py in <cell line: 9>()
    141
    142         elif nilm["model"] == "VAE":
--> 143             history = model.fit((x_train-main_mean)/main_std, (y_
ain-app_mean)/app_std, validation_split=nilm["training"]["validation_spli
t"], shuffle=True,
    144                                 epochs=epochs, batch_size=batch_s
e, callbacks=list_callbacks, verbose=1, initial_epoch=0)
    145

~/anaconda3/envs/tf/lib/python3.10/site-packages/keras/utils/traceback_ut
s.py in error_handler(*args, **kwargs)
     68             # To get the full stack trace, call:
     69             # `tf.debugging.disable_traceback_filtering()`
---> 70             raise e.with_traceback(filtered_tb) from None
     71         finally:
     72             del filtered_tb

~/anaconda3/envs/tf/lib/python3.10/site-packages/keras/engine/training.py
n tf__train_function(iterator)
     13                 try:
     14                     do_return = True
---> 15                     retval_ = ag__.converted_call(ag__.ld(step_fu
tion), (ag__.ld(self), ag__.ld(iterator)), None, fscope)
     16                 except:
     17                     do_return = False

ValueError: in user code:

    File "/home/mariana2/anaconda3/envs/tf/lib/python3.10/site-packages/k
as/engine/training.py", line 1160, in train_function  *
        return step_function(self, iterator)
    File "/home/mariana2/anaconda3/envs/tf/lib/python3.10/site-packages/k
as/engine/training.py", line 1146, in step_function  **
        outputs = model.distribute_strategy.run(run_step, args=(data,))
    File "/home/mariana2/anaconda3/envs/tf/lib/python3.10/site-packages/k
as/engine/training.py", line 1135, in run_step  **
        outputs = model.train_step(data)
    File "/home/mariana2/anaconda3/envs/tf/lib/python3.10/site-packages/k
as/engine/training.py", line 993, in train_step
        y_pred = self(x, training=True)
    File "/home/mariana2/anaconda3/envs/tf/lib/python3.10/site-packages/k
as/utils/traceback_utils.py", line 70, in error_handler
        raise e.with_traceback(filtered_tb) from None
    File "/home/mariana2/anaconda3/envs/tf/lib/python3.10/site-packages/k
as/engine/input_spec.py", line 216, in assert_input_compatibility
        raise ValueError(

    ValueError: Layer "model" expects 2 input(s), but it received 1 input
ensors. Inputs received: [<tf.Tensor 'IteratorGetNext:0' shape=(None, 102
1) dtype=float32>]
```

In [ ]:
```
conda install nbconvert[webpdf]
```

```
Collecting package metadata (current_repodata.json): /
```

In [ ]:
```

```