Proposal

Two-steps:

- Utilizing random forest to filter features
- Training a 1D CNN and transformer combined model

Data Utilization

Training data: training-data_c and training-data_a

Test data-1: split from training data

Test data: test-data_c

Result (Random Forest, test data-1)

Accuracy: 0.94397

Classification Report:

Classification Report:

	precision	recall	f1-score	support
amfx1_memory-stress-start	0.81	0.68	0.74	19
ausfx1_vcpu-overload-start	0.95	1.00	0.98	21
normal	1.00	0.54	0.70	13
amfx1_bridge-delif	1.00	1.00	1.00	21
amfx1_ens5_interface-loss-start-70	1.00	1.00	1.00	12
ausfx1_ens4_interface-down	0.75	0.32	0.44	19
udmx1_ens4_interface-loss-start-70	1.00	0.94	0.97	18
ausfx1_bridge-delif	0.92	0.73	0.81	15
ausfx1_memory-stress-start	1.00	1.00	1.00	15
ausfx1_ens4_interface-loss-start-70	1.00	1.00	1.00	19
udmx1_vcpu-overload-start	0.94	0.99	0.97	560
udmx1_ens4_interface-down	0.90	0.50	0.64	18
amfx1_ens5_interface-down	1.00	1.00	1.00	11
udmx1_bridge-delif	0.93	0.78	0.85	18
udmx1_memory-stress-start	1.00	1.00	1.00	17
amfx1_vcpu-overload-start	1.00	1.00	1.00	25
accuracy			0.94	821
macro avg	0.95	0.84	0.88	821
weighted avg	0.94	0.94	0.94	821

Result (Random Forest, test data-1)

Confusion Matrix:

```
Confusion Matrix:
 [[ 13   0   0   0   0   1   0   0   0   0   5   0   0   0   0   0]
 [  0  21   0   0   0   0   0   0   0   0   0   0   0   0   0   0]
 [  0   0   7   0   0   0   0   0   0   0   6   0   0   0   0   0]
 [  0   0   0  21   0   0   0   0   0   0   0   0   0   0   0   0]
 [  0   0   0   0  12   0   0   0   0   0   0   0   0   0   0   0]
 [  1   1   0   0   0   6   0   1   0   0  10   0   0   0   0   0]
 [  0   0   0   0   0   0  17   0   0   0   1   0   0   0   0   0]
 [  0   0   0   0   0   0   0  11   0   0   3   1   0   0   0   0]
 [  0   0   0   0   0   0   0   0  15   0   0   0   0   0   0   0]
 [  0   0   0   0   0   0   0   0   0  19   0   0   0   0   0   0]
 [  2   0   0   0   0   0   0   0   0   0 557   0   0   1   0   0]
 [  0   0   0   0   0   1   0   0   0   0   8   9   0   0   0   0]
 [  0   0   0   0   0   0   0   0   0   0   0   0  11   0   0   0]
 [  0   0   0   0   0   0   0   0   0   0   4   0   0  14   0   0]
 [  0   0   0   0   0   0   0   0   0   0   0   0   0   0  17   0]
 [  0   0   0   0   0   0   0   0   0   0   0   0   0   0   0  25]]
```

Result (Random Forest, test data-1)

Confusion Matrix:

Result (Random Forest, test data-1)

Accuracy: 0.84307

Classification Report:

Classification Report:

	precision	recall	f1-score	support
normal	0.36	0.26	0.30	19
udmx1_vcpu-overload-start	0.20	1.00	0.34	19
amfx1_ens5_interface-loss-start-70	1.00	0.58	0.73	19
ausfx1_vcpu-overload-start	1.00	1.00	1.00	19
ausfx1_ens4_interface-loss-start-70	1.00	1.00	1.00	19
udmx1_ens4_interface-down	0.25	0.11	0.15	19
ausfx1_bridge-delif	1.00	1.00	1.00	19
ausfx1_ens4_interface-down	0.75	0.47	0.58	19
amfx1_bridge-delif	1.00	1.00	1.00	19
udmx1_bridge-delif	1.00	1.00	1.00	19
amfx1_memory-stress-start	0.92	0.89	0.91	588
udmx1_memory-stress-start	0.83	0.26	0.40	19
amfx1_ens5_interface-down	0.95	1.00	0.97	19
udmx1_ens4_interface-loss-start-70	1.00	0.53	0.69	19
ausfx1_memory-stress-start	1.00	1.00	1.00	19
amfx1_vcpu-overload-start	1.00	1.00	1.00	19
accuracy			0.84	873
macro avg	0.83	0.76	0.75	873
weighted avg	0.89	0.84	0.85	873

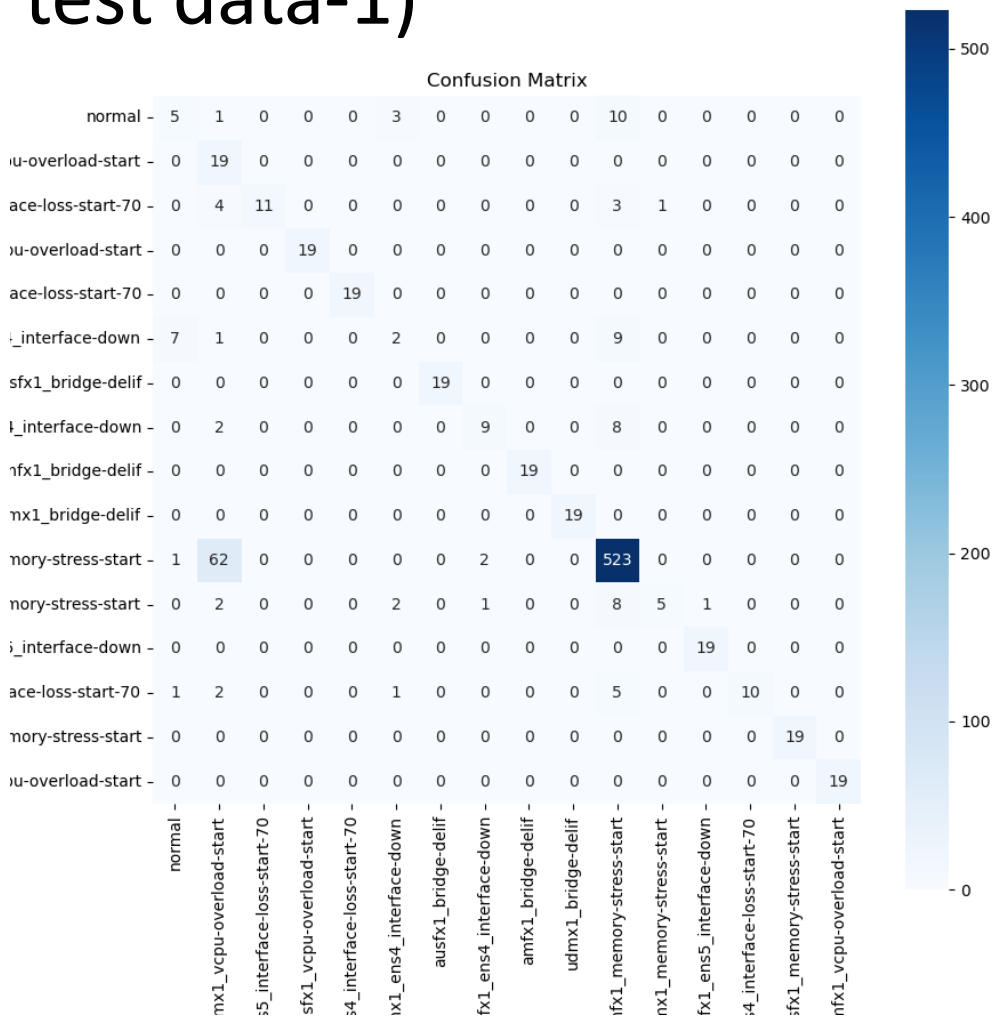
Result (Random Forest, test data-1)

Confusion Matrix:

```
Confusion Matrix:
 [[  5   1   0   0   0   3   0   0   0   0  10   0   0   0   0   0]
 [  0  19   0   0   0   0   0   0   0   0   0   0   0   0   0   0]
 [  0   4  11   0   0   0   0   0   0   0   3   1   0   0   0   0]
 [  0   0   0  19   0   0   0   0   0   0   0   0   0   0   0   0]
 [  0   0   0   0  19   0   0   0   0   0   0   0   0   0   0   0]
 [  7   1   0   0   0   2   0   0   0   0   9   0   0   0   0   0]
 [  0   0   0   0   0   0  19   0   0   0   0   0   0   0   0   0]
 [  0   2   0   0   0   0   0   9   0   0   8   0   0   0   0   0]
 [  0   0   0   0   0   0   0   0  19   0   0   0   0   0   0   0]
 [  0   0   0   0   0   0   0   0   0  19   0   0   0   0   0   0]
 [  1  62   0   0   0   0   0   2   0   0 523   0   0   0   0   0]
 [  0   2   0   0   0   2   0   1   0   0   8   5   1   0   0   0]
 [  0   0   0   0   0   0   0   0   0   0   0   0  19   0   0   0]
 [  1   2   0   0   0   1   0   0   0   0   5   0   0  10   0   0]
 [  0   0   0   0   0   0   0   0   0   0   0   0   0   0  19   0]
 [  0   0   0   0   0   0   0   0   0   0   0   0   0   0   0  19]]
```

Result (Random Forest, test data-1)

Confusion Matrix:



Result CNN-Transformer Model

I apologize for unable to provide any result of it.

Since my GPU run out of memory during test after training

I will show the model structure on the next page

Model Structure

Model: "cnn_transformer_model"

Layer (type)	Output Shape	Param #
conv1d (Conv1D)	multiple	2
conv1d_1 (Conv1D)	multiple	6
conv1d_2 (Conv1D)	multiple	36
conv1d_3 (Conv1D)	multiple	80
conv1d_4 (Conv1D)	multiple	1056
conv1d_5 (Conv1D)	multiple	8256
normalization (Normalization)	multiple	3
max_pooling1d (MaxPooling1D)	multiple	0
transformer (Transformer)	multiple	9443
average_pooling1d (AveragePooling1D)	multiple	0
conv1d_6 (Conv1D)	multiple	36992
conv1d_7 (Conv1D)	multiple	98560
normalization_1 (Normalization)	multiple	193
flatten (Flatten)	multiple	0
dense_2 (Dense)	multiple	163904
dense_3 (Dense)	multiple	2080
dropout_2 (Dropout)	multiple	0
dense_4 (Dense)	multiple	528

Total params: 321,139
Trainable params: 320,943
Non-trainable params: 196

Next Step

Consider to use GAN to generate data similar with domain C before training

In order to increase the similarity of data come from mirror A

Thanks for your evaluation

Name: Jiaxing LU
E-mail: jiaxinglu0118@gmail.com
Date: 2023.10.31