Fairseq-Signals Results:

a. Summary of the paper:

- ECG is a widely used method for assessing heart health without invasive equipment.
- Deep learning models have been used to address heart issues like arrhythmia and myocardial infarction using ECG data.
- Two main challenges: limited labeled data and difficulties in adapting models to different ECG leads.
- A proposed method focuses on generalizing ECG data regardless of lead combinations.
- The paper introduces two key contributions: local and global contextualization of raw ECG inputs and masking tokens in the input stage.
- The method improves local representation power with a module called "wav2vec" and enhances global context with "Contrastive Multilead Self-Supervised Learning (CMSC)".
- The network optimization involves using two loss functions, one for masked time steps and another for global features.

b. Setup:

- The transformer model is pretrained on the Physionet2021 and PTB-XL dataset by the proposed policy which is referred as wave2vec-cmsc-rlm(random lead masking).
- Then the pretrained model is finetuned on cardiac arrhythmia classification and patient identification tasks.
- The github repository here with few changes in the code was used for both pretraining and finetuning.
- The repository provides with preset configurations for the both tasks. The changes in pretraining are mentioned on the next page.
- For finetuning except for leads to load(kept as null), every other parameter was kept same as the pretrain config.

Commands:

Training:

fairseq-hydra-train task.data=/absolute/path/to/manifest/cmsc --config-dir /path/to/examples/w2v_cmsc/config/pretraining --config-name w2v_cmsc_rlm

Finetuning:

fairseq-hydra-train task.data=/absolute/path/to/manifest_tuning/finetune model.model_path=/absolute/path/to/checkpoint criterion.report_cinc_score=True criterion.weights_file=/absolute/path/to/weights.csv --config-dir path/to/examples/w2v_cmsc/config/finetuning/ecg_transformer --config-name diagnosis

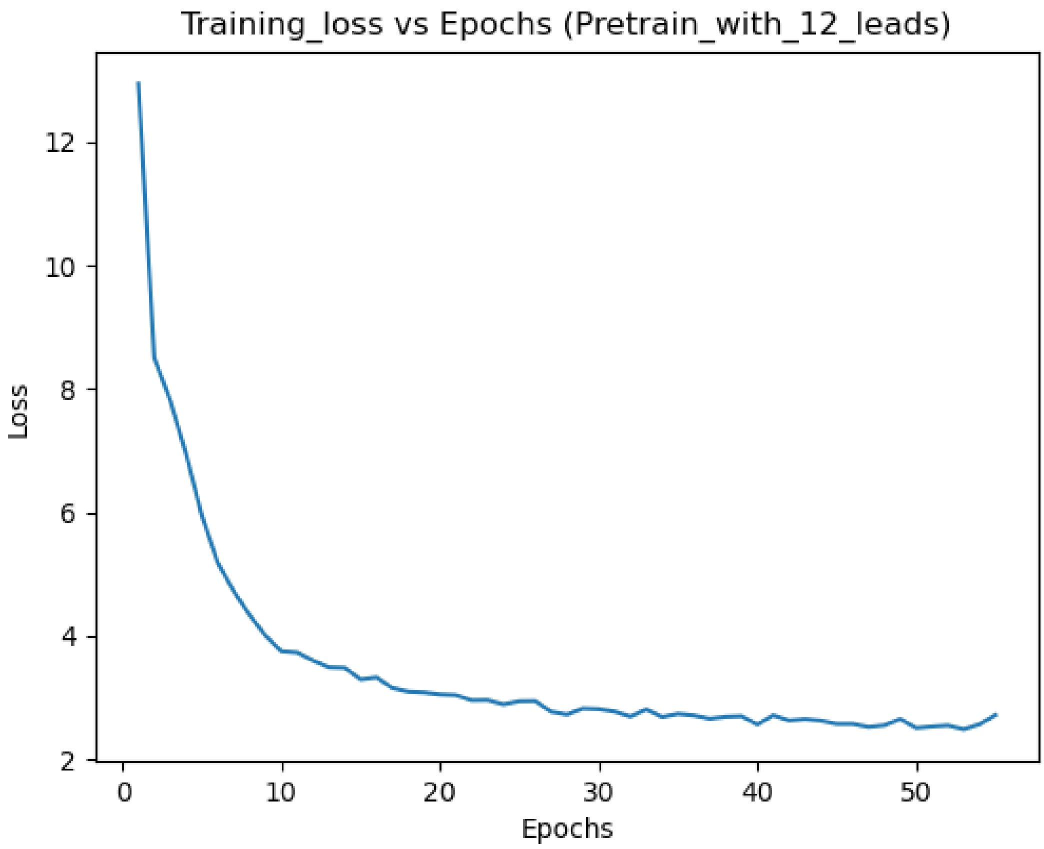
Pretrain Config

```yaml
task:
  _name: ecg_pretraining
  data: ???
  perturbation_mode: ["random_leads_masking"]
  p: [1.0]
  mask_leads_selection: random
  mask_leads_prob: 0.5
  normalize: false
  enable_padding: true
  enable_padding_leads: true # Changed from false
  leads_to_load: '[0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11]' # Changed from null

dataset:
  num_workers: 1  # Changed from 6
  max_tokens: null
  batch_size: 16 # Changed from 128
  valid_subset: ""
  disable_validation: true

distributed_training:
  distributed_world_size: 1  # Changed from 4

criterion:
  _name: wav2vec2_with_cmsc
  infonce: true
  log_keys: ["prob_perplexity", "code_perplexity", "temp"]
  loss_weights: [0.1, 10]

optimization:
  max_epoch: 55  # Changed from 200
  lr: [5e-5]
  update_freq: [2]
```

c. Results:

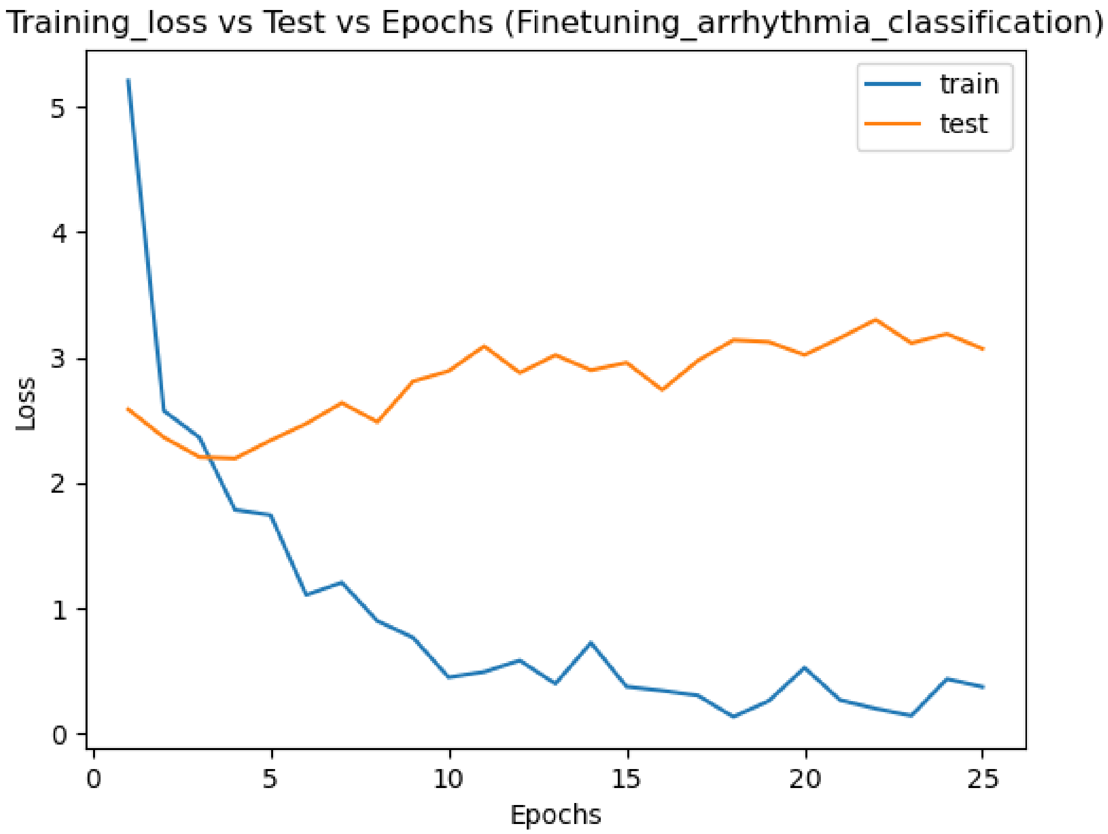
1. Pretrain results:

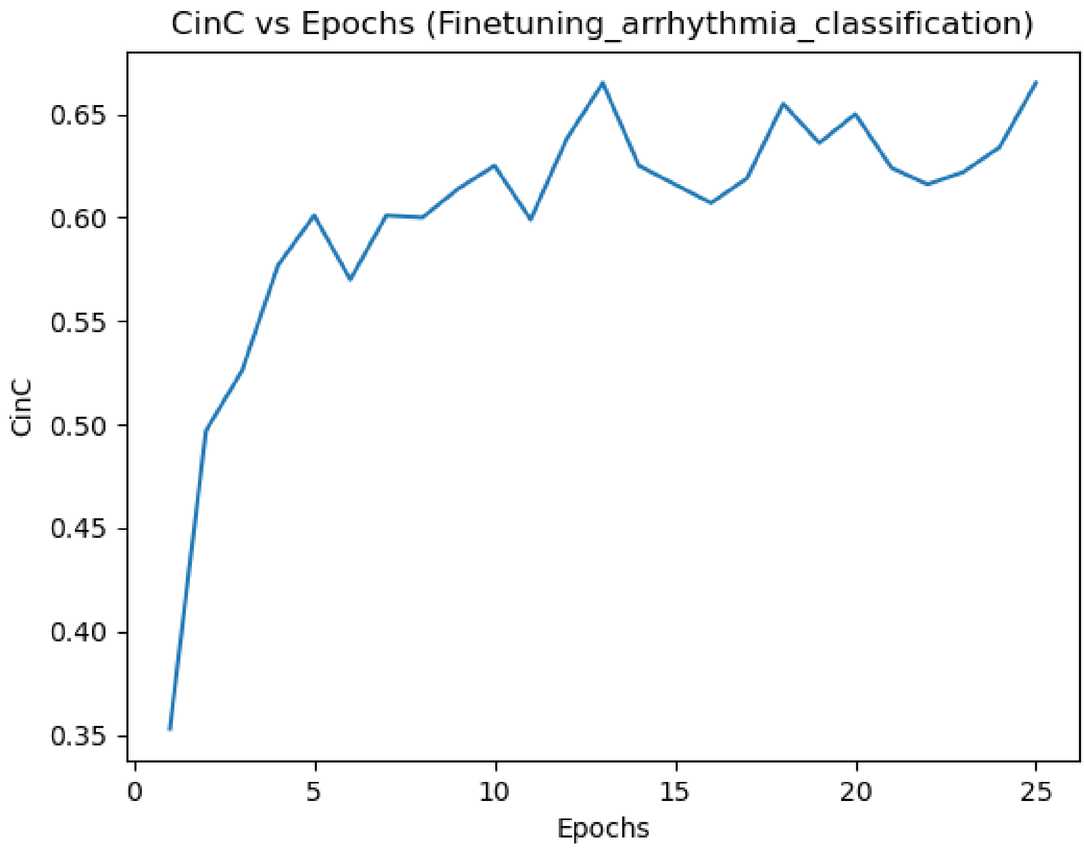
- As mentioned in the config file above, the model was trained until 55 epochs were completed.
- The training loss was equal to 2.552 with accuracy close to 61% on train split.
- The trajectory of the loss can be seen below in the “epoch vs loss” plot.



2. Finetune results:

- The checkpoint model that was trained for 55 epochs was used as the backbone for finetuning on arrhythmia classification task.
- It was finetuned for 25 epochs, and the loss was equal to 3.072 with CinC score equal to 0.665 on the test set.
- The results are summarized below with plots.


Training_loss vs Test vs Epochs (Finetuning_arrhythmia_classification)
train
test
Loss
Epochs
0
1
2
3
4
5
0
5
10
15
20
25


CinC vs Epochs (Finetuning_arrhythmia_classification)
CinC
Epochs
0.35
0.40
0.45
0.50
0.55
0.60
0.65
0
5
10
15
20
25

d. Comments:

- The test loss stays moderately flat and marginally increases in the end. This can be a case of overfitting.
- The CinC score also increases, this maybe, because the model is performing on majority of the test datapoints, and the increase in loss is due to the minority ones.
- The paper reports a CinC score of 0.732 ± 0.004 on arrhythmia classification task but here the score is 0.665 which is 0.06 units below the former value.
- The exact reasons need to be researched further.