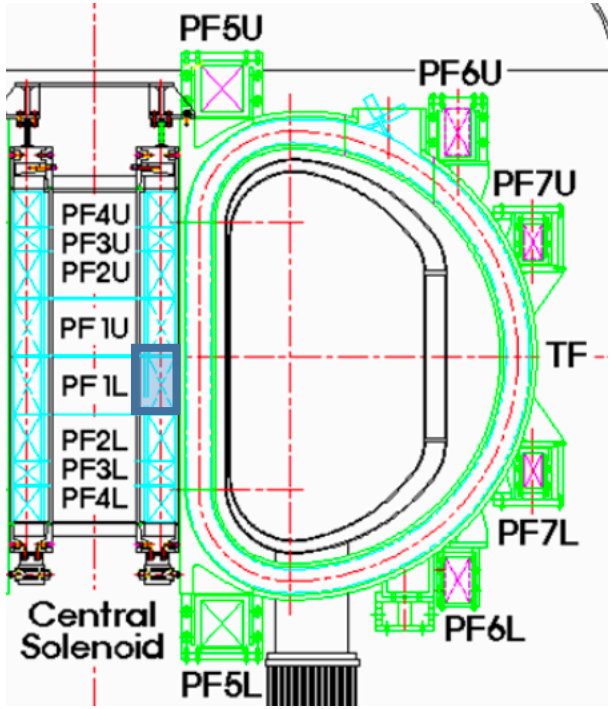
Time series PF coil temperature forecasting using recurrent transformer model.

Giil Kwon, Taehyun Tak, Hyunjung Lee

Introduction

❖KSTAR superconducting coil system

- The superconducting coil system is one of the most important components in Korea Superconducting Tokamak Advanced Research (KSTAR), which has been operated since 2008.
- Superconducting coils are cooled by forced-flow supercritical helium about 4.5 K.
- To protect the superconducting coil system, we need to predict next superconducting coil temperature.
- The rising of poloidal field (PF) coil's temperature is mostly by AC losses according to the variations of current and magnetic field

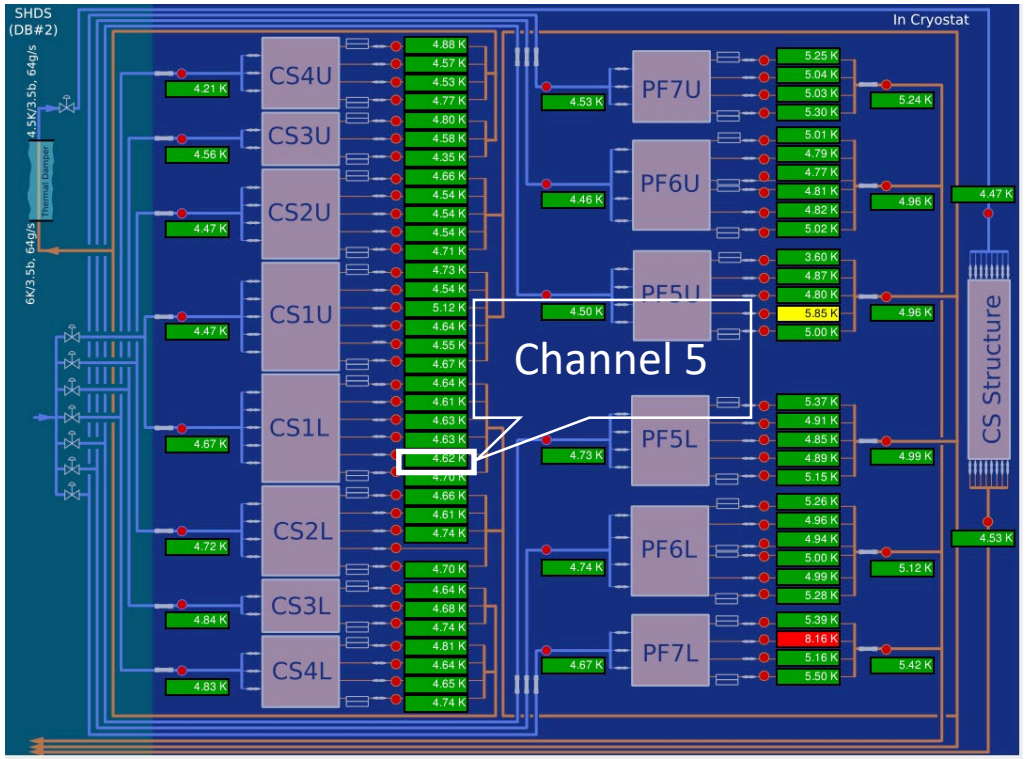


KSTAR Tokamak Images

Introduction

❖The measurement of the coil temperature

- Coil's temperature are measured on Tokamak Monitoring System (TMS).
  - TMS measured the temperature data and publish PV every 1 second.
- Thermometer measure the temperature of coolant(Helium).
  - Thermometers are located at inlet and outlet of each cooling channels.
  - In this work, we use data from outlet channel 5 thermometer of the PF1 coil.



Tokamak Monitoring System(TMS) GUI

Introduction

❖Problem formulation

- Suppose we have a collection of N univariate time series data, $\{X_{n,t_0:t_j}\}_{n=1}^{N}$, where $X_{n,t_0:t_j} = \left[X_{n,t_0}, X_{n,t_1}, \cdots, X_{n,t_j}\right], X_{n,t} \in \mathbb{R}$, denotes $n'th$ time series data value at time t. we will predict the next τ step time series values, $\{X_{n,t_{j+1}:t_{j+\tau}}\}_{n=1}^{N}$.
  - In this work, N=1, τ=1, we will do one step ahead forecast of univariate time series data.
  - we will forecast the temperature of PF1L(channel 5) data.
- We are going to formulate the multi step prediction into one-step ahead prediction problem, where Φ is learnable parameter while training.

$$P\left(X_{n,t_i+\Delta+1:t_i+\Delta+\tau+1} \middle| X_{n,t_i:t_i+\Delta}; \Phi\right) = \prod_{t=t_i}^{t_i+\tau} P(X_{n,t+\Delta+1} | X_{n,t:t+\Delta}; \Phi)$$

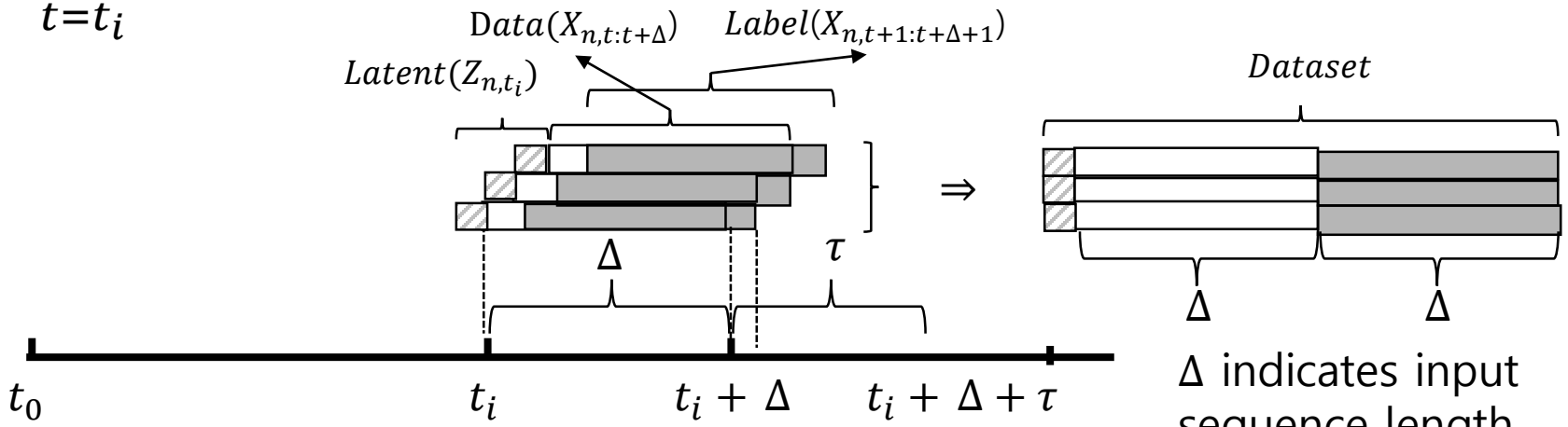
Introduction

❖Problem formulation

- We change this model to recurrent model by adding latent variable Z_{n,t_i}.
- Z_{n,t_i} denotes the latent variable of n'th time series data at time t_i.

$$P\left(X_{n,t_i+\Delta+1:t_i+\Delta+\tau+1} \middle| X_{n,t_i:t_i+\Delta}; \Phi'\right)$$

$$= \prod_{t=t_i}^{t_i+\tau} P(X_{n,t+1:t+\Delta+1}, Z_{n,t_i+1} | X_{n,t:t+\Delta}, Z_{n,t_i}; \Phi')$$



Δ indicates input sequence length

Model Architecture

❖Long-Short-Term Memory(LSTM)

- LSTM is one kind of Recurrent neural network.
- LSTM is well suited to predict, classify the time series data.

❖Transformer

- Transformer is the self-attention based deep learning model.
- It mostly used to solve Natural Language Problem(NLP).(such as GPT-3, BERT).
- Nowadays, it also used to forecast time series data.(Informer, Perceiver, Reformer)

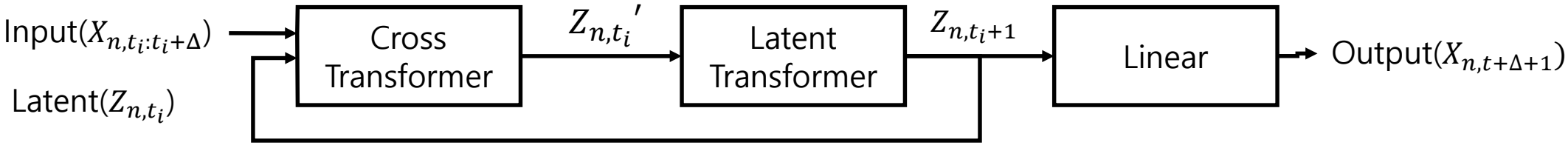
❖Perceiver

- The model build upon the Transformer that can learn multimodal data.
- The model iteratively attend to the input data by alternating cross-attention and latent transformer.
- The model unrolled in depth to the same input rather then in time to different inputs

Model Architecture

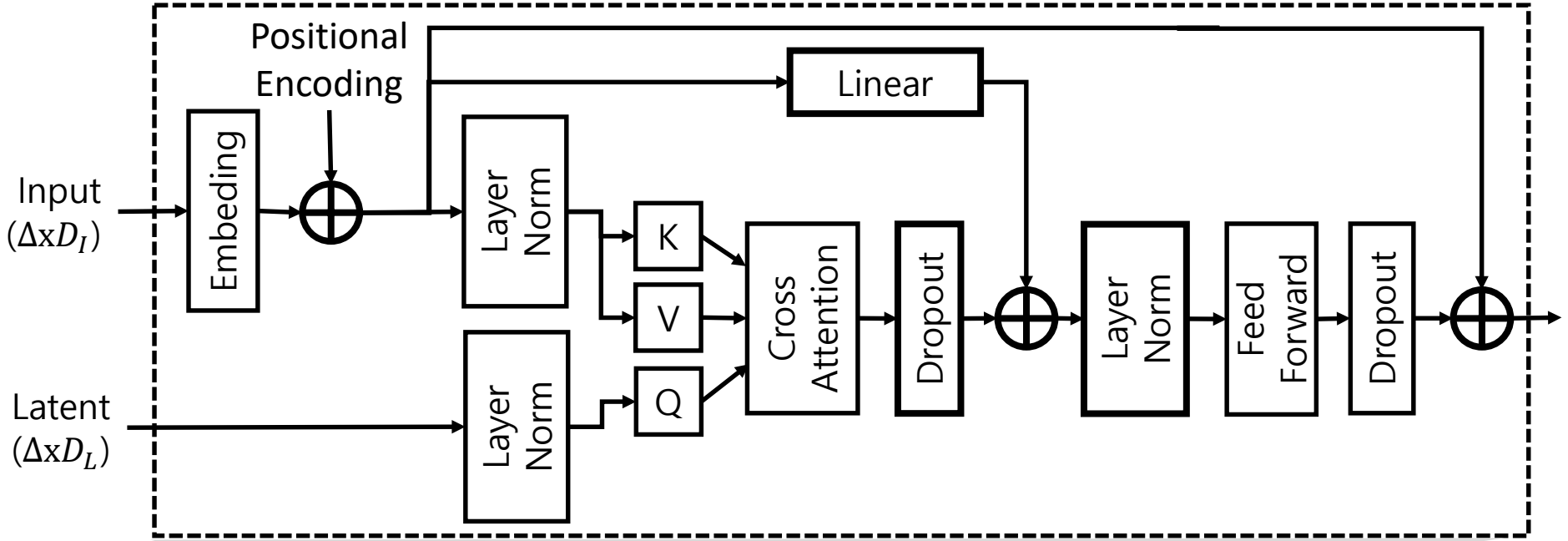
❖Recurrent Transformer(RT)

- We build recurrent transformer upon the perceiver by iteratively giving time series input and attend input data to latent variable by using cross-transformer module.
- As the perceiver did, our model also alternating cross-attend module and latent transformer.
- As RNN did, we give latent variable as input to the model.
- Unlike Perceiver, our model unrolled in time to different inputs.



Model Architecture

❖ Recurrent Transformer

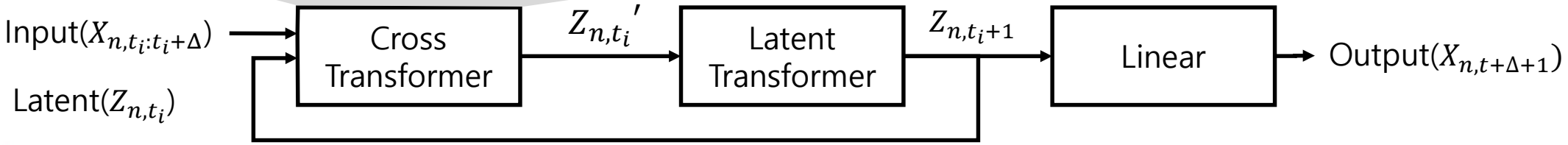


Cross Attention

$$Cross\ Attention(Q', K, V) = softmax\left(\frac{Q'K^T}{\sqrt{d_k}}\right)V,$$

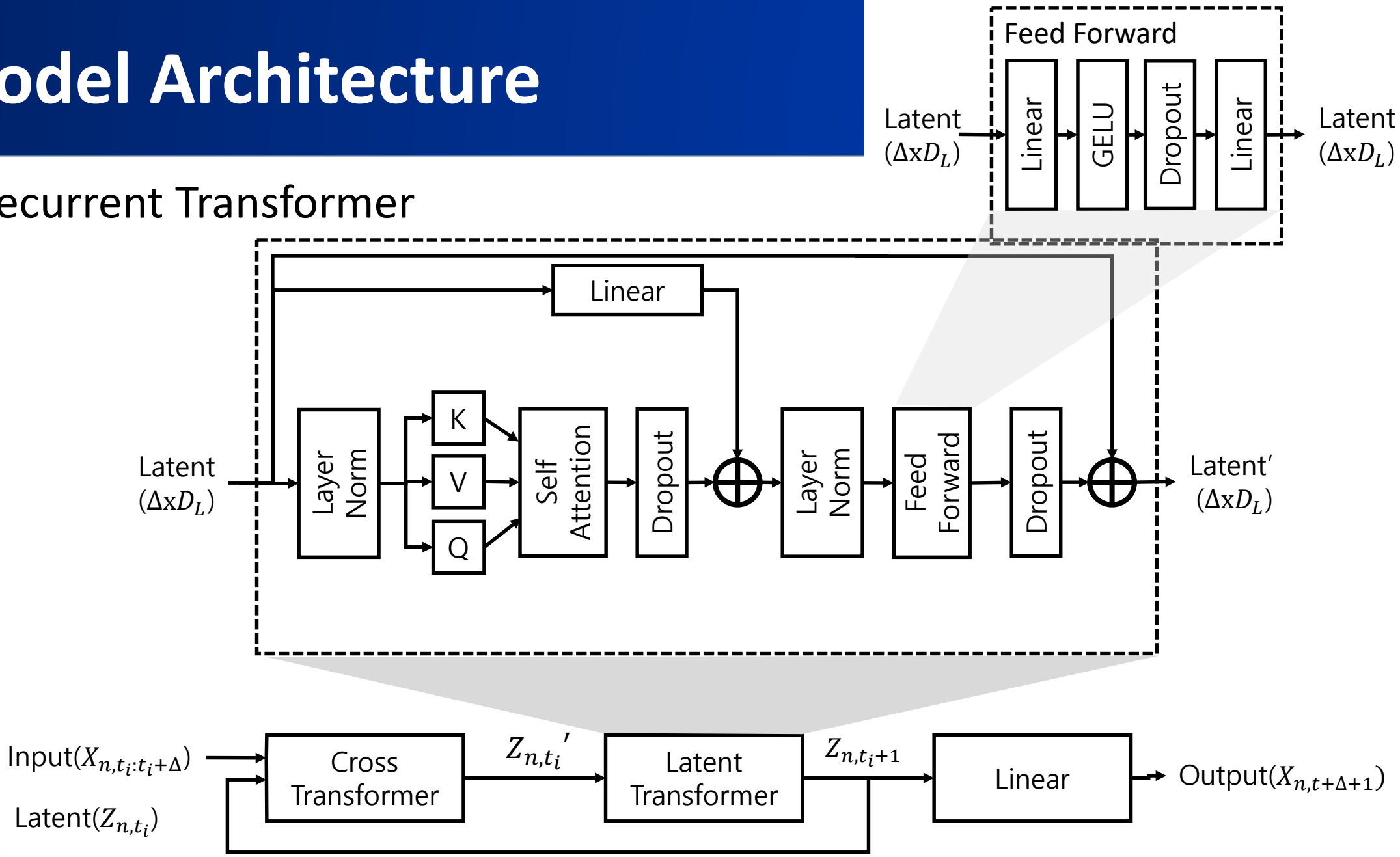
$where\ Q' \in \mathbb{R}^{\Delta x D_L}, K \in \mathbb{R}^{\Delta x D_I}\ and\ V \in \mathbb{R}^{\Delta x D_I}$

D_I denotes dimension of input data(1).
D_L indicates dimension of latent variable(48).
$D_I \ll D_L$



Model Architecture

❖Recurrent Transformer

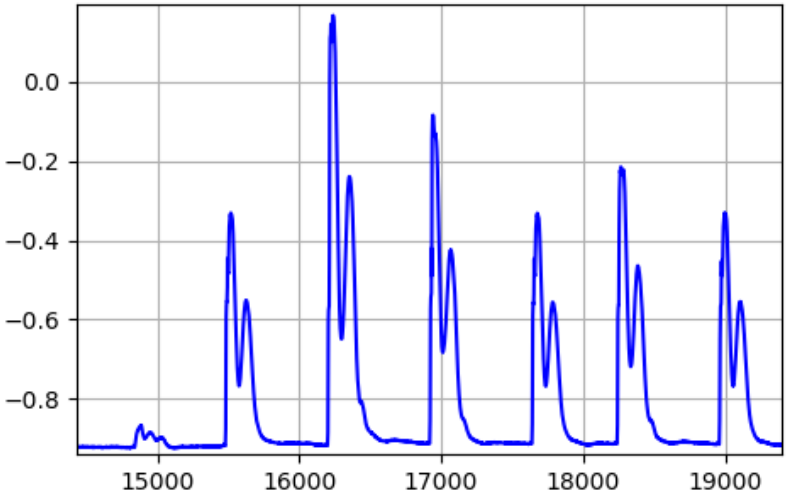


Experiment

❖KSTAR Dataset

- Data acquired from Magnet Power Supply(MPS), Helium Distribution System(HDS), Tokamak Monitoring System(TMS).
  - MPS generate PF current related PV, TMS generate temperature related PV, HDS generate helium pressure and flow rate related PV.
- We have interpolated the data to have each data every 1 seconds. and remove noise.
- It consists of data collected between 2018/11/01 an 2018/11/30(633shots,24days, 820800 samples)
  - Train data(19 days,615600 samples) : 79%
  - Test data(5days, 171000 samples): 21%

Example Data



Experiment

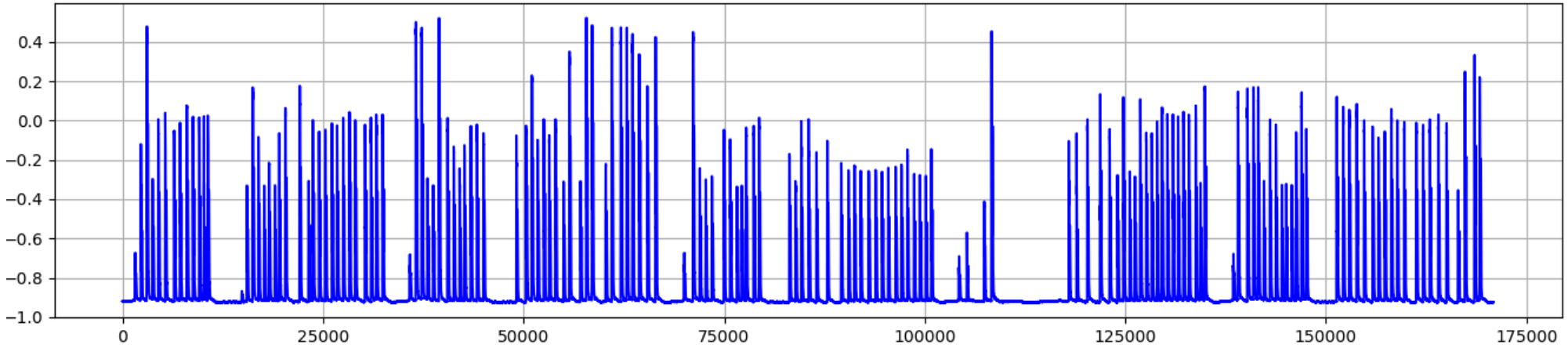
❖Environment

- We use Pytorch library to build our program.
- We use 1 GPU card when we training the dataset.

Hardware specification

	Name	# of HW
CPU	AMD Ryzen Threadripper 1920x	2
GPU	Nvidia 2070 super	1

Dataset



Experiment

❖Training

- We used Adam W optimizer with $\beta_1 = 0.9, \beta_2 = 0.99\ and\ \varepsilon = 10^{-8}$. And StepLR scheduler with step size = 1.0 and $\gamma = 0.9$.
- Model dimension = 48 and latent dimension = 48
- Training Time : it takes 20 hour to train the Recurrent Transformer that uses 100 length sequence as input(256 batch).

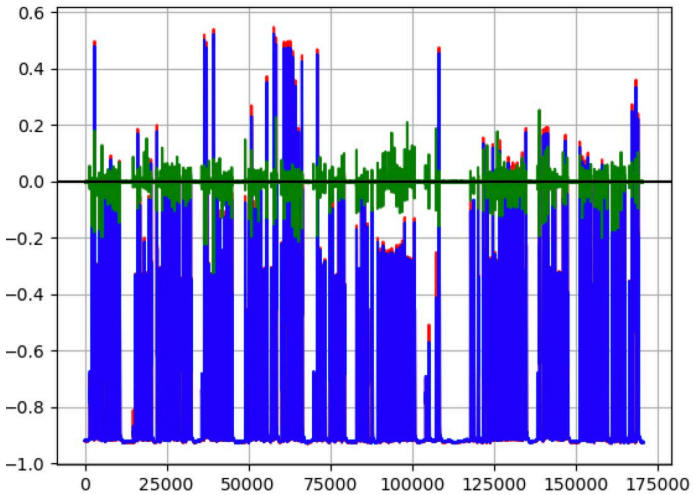
❖Model Parameter Setting

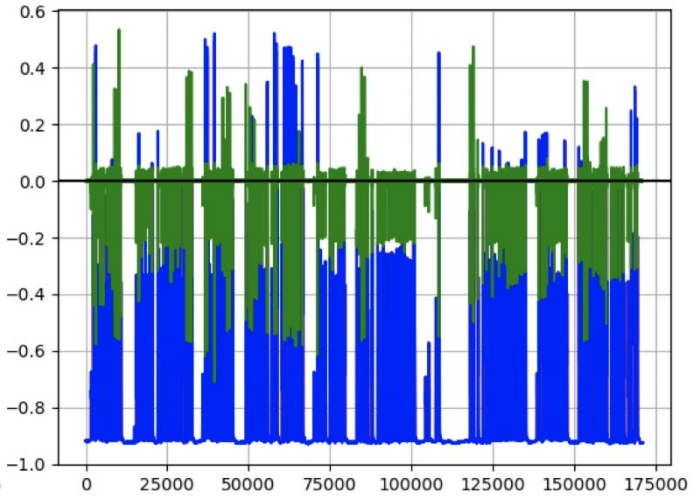
- LSTM : hidden size = 96, # of layer = 1, Epoch = 200, batch size = 256, input window = 100, input dimension = 47
- Transformer : # of head = 8, # of layer = 2, Epoch = 200, batch size = 256, input window = 100, input dimension = 48
- Recurrent Transformer: # of head = 8, # of layer = 1, Epoch = 200, batch size = 256, input window = 100, input dimension = 48

Experiment

❖Forecast graph (total test set result)

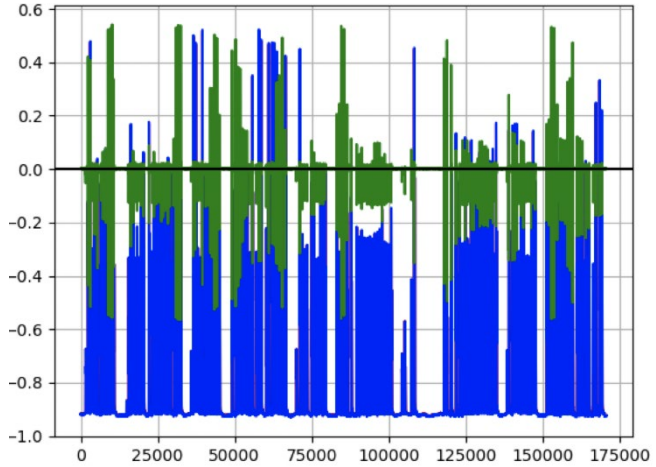
Transformer

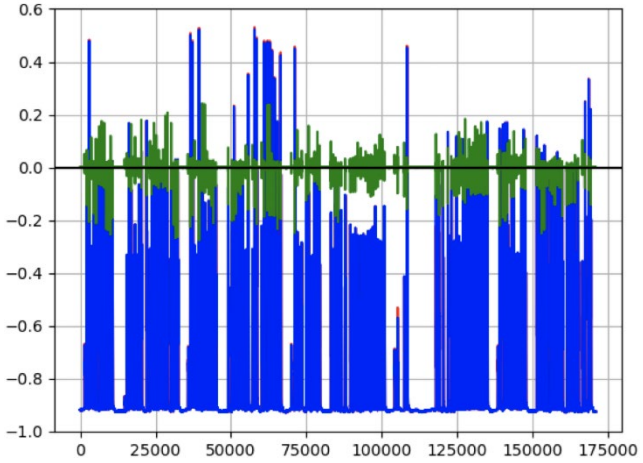




LSTM

Recurrent Transformer (seq1)



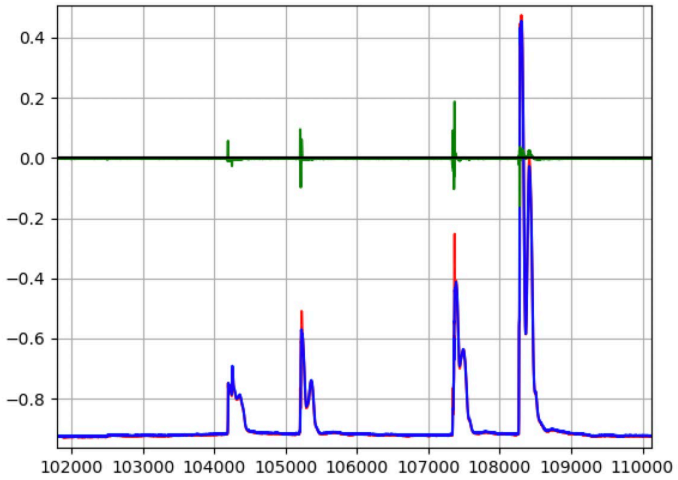


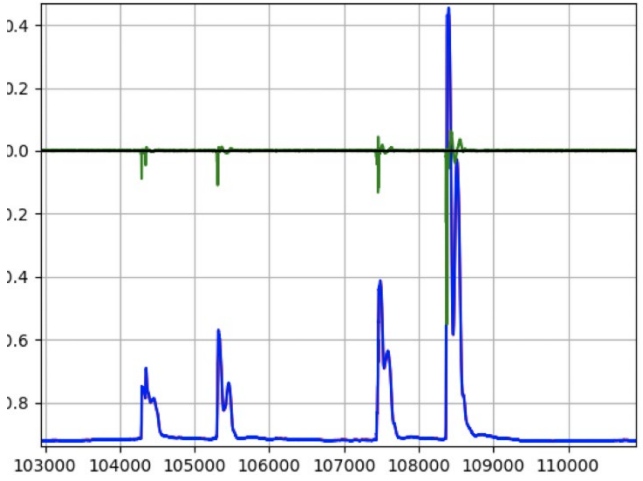
Recurrent Transformer (seq100)

Experiment

❖Forecast graph(one shot result)

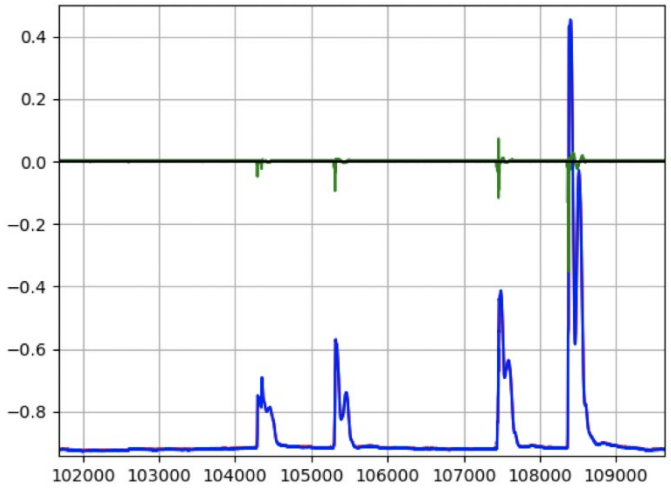
Transformer

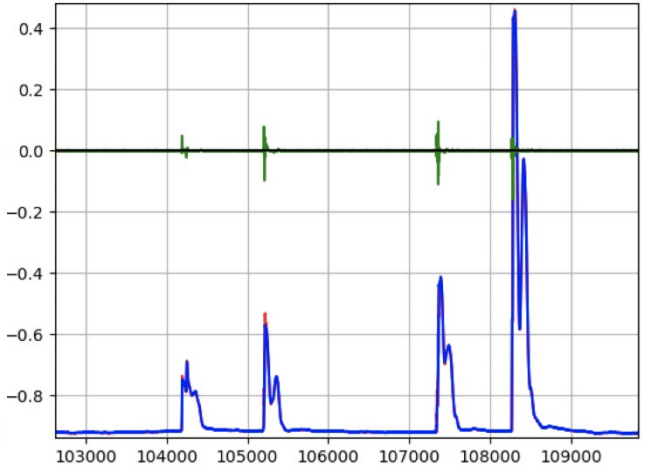




LSTM

Recurrent Transformer (seq1)



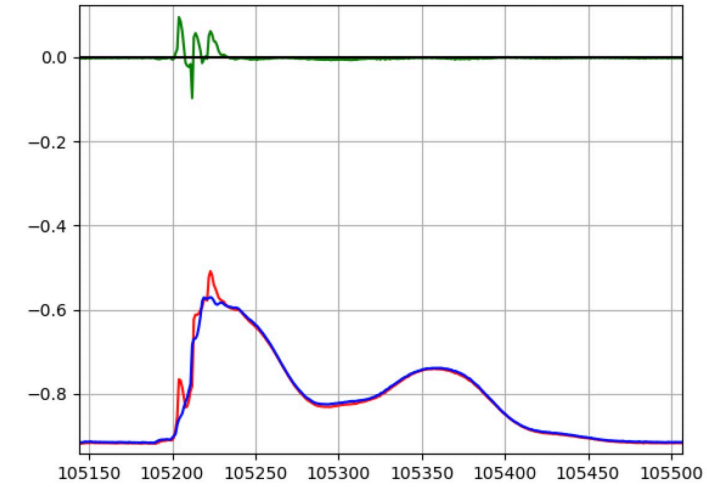


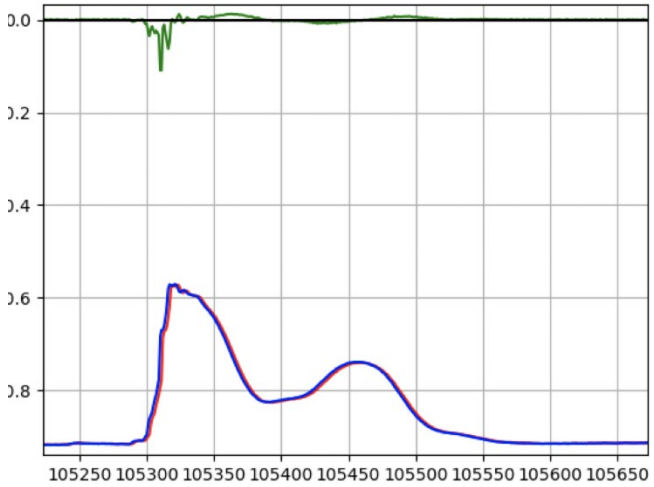
Recurrent Transformer (seq100)

Experiment

❖Forecast graph(one shot result)

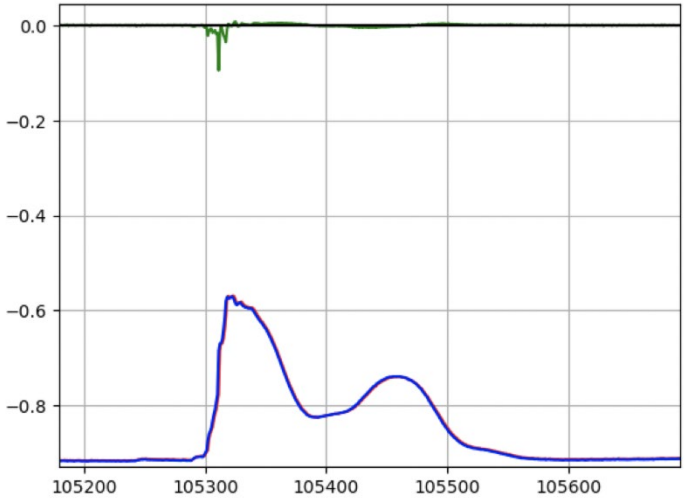
Transformer

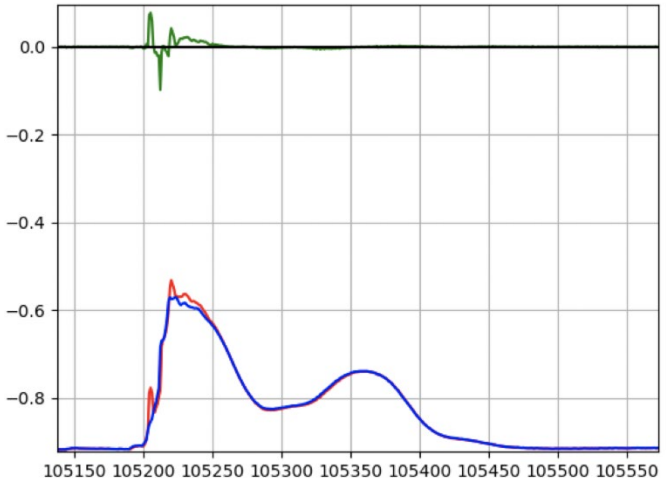




LSTM

Recurrent Transformer (seq1)





Recurrent Transformer (seq100)

Experiment

❖One step ahead forecast test result

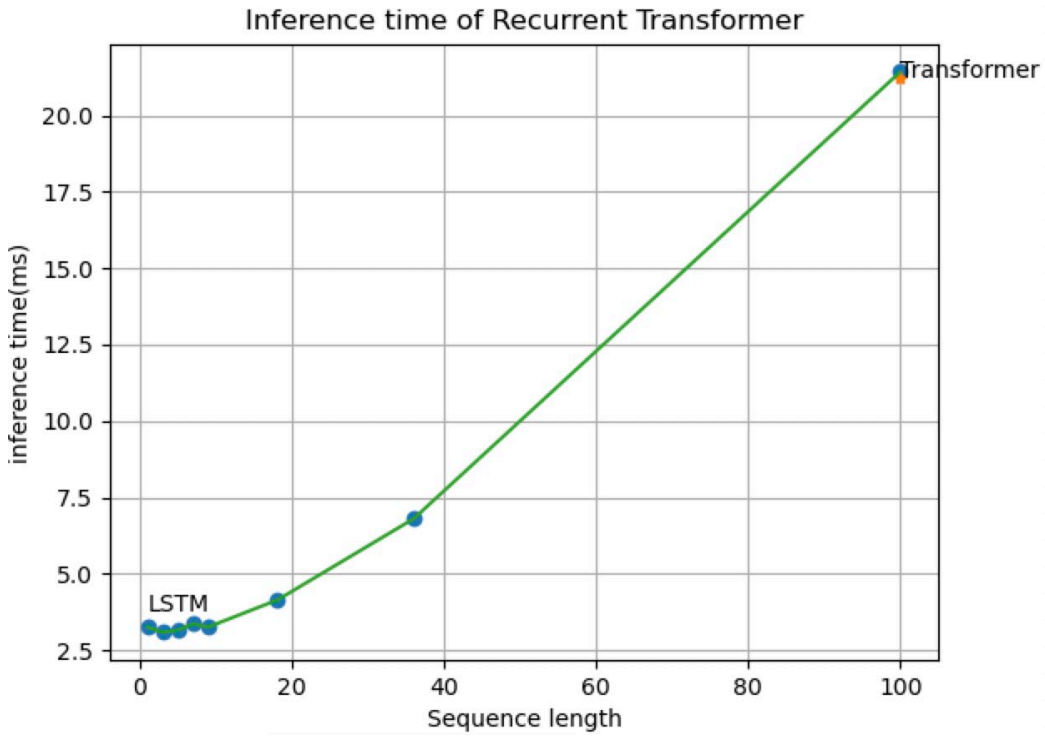
- We compare forecast accuracy of the Recursive Transformer(RT) with other algorithms(LSTM, Transformer) which has similar parameters(batch size=256, model dimension = 48, # of epoch = 200).
  - RT with sequence length 100(seq100) is more accurate than others
  - RT with sequence length 1(seq1) also has similar forecast accuracy to the conventional Transformer.

	R2 score	MAE	RMSE	MAPE	Quantile score(P10/P50/P90)
LSTM	0.9916	0.0048	0.0198	0.0377	(0.0024/0.0024/0.0024)
Transformer(2layer, seq 100)	0.9992	0.0029	0.0061	0.0830	(0.0019/0.0014/0.0009)
Recursive Transformer(seq1)	0.9973	0.0017	0.0111	0.0258	(0.0009/0.0008/0.0007)
Recursive Transformer(seq64)	0.9988	0.0018	0.0071	0.0892	(0.0014/0.0009/0.0003)
Recursive Transformer(seq100)	**0.9992**	**0.0013**	**0.0061**	**0.0112**	**(0.0009/0.0006/0.0003)**

Experiment

❖Inference time

- We measured one sample inference time of models.
- The inference time of RT when seq1 is about 3.2541 ms.
- The inference time of **RT** when sequence length 100 is about 21.433377ms.
- The inference time of **Transformer** when sequence length 100 is about 19.78277ms
- The inference time of **RT** when sequence length 100 is slightly larger that the transformer inference time.



Inference time of Recurrent Transformer

Conclusion

❖We have presented the Recurrent Transformer, a perceiver based model that can one step ahead forecast of univariate time series data.

- Our model successfully learn and one step ahead forecast of KSTAR PF1 temperature data.
- Our model predicted time series data more accurately than other existing models(LSTM, Transformer).

❖In the future, we would like to forecast multi step ahead forecast using univariate data or using multi variate data.

The End

Appendix

Introduction

❖Why we choose channel 5 of PF 1 temperature sensor

- To measure the temperature variation caused by AC loss, we chose the channel 5.
- because Channel 5 is most affected by magnetic field

=>This channel mostly affected by operation of PF coil

❖Why temperature of PF1 coil

- We have already recognized that PF1 coil shows a larger temperature rise compared to other PF coils and it will become one of important factors for KSTAR PF coil operation.

Experiment

❖Performance Metrics

- R2-score
  - Suppose $y_1, y_2, \cdots, y_n$ denote dataset values, $f_1, f_2, \cdots, f_n$ denote predicted values And $\overline{y_1} = \frac{1}{n}\sum_{i=1}^{n} y_i$
  - $R^2 = 1 - \frac{\sum_i (y_i - \bar{y})^2}{\sum_i (y_i - f_i)^2}$
  - Best score is 1.0 and worst score is 0.0
- Mean Absolute Error(MAE)
  - $MAE = \frac{\sum_{i=1}^{n} (y_i - f_i)^2}{n}$
- Root Mean Square E(RMSE)
  - $RMSE = \sqrt{MSE} = \sqrt{\frac{\sum_{i=1}^{n} (y_i - f_i)^2}{n}}$
- Mean Absolute Percentage Error(MAPE)
  - $MAPE = \frac{100}{n}\sum_{i=1}^{n} \left|\frac{y_i - f_i}{y_i}\right|$
- Quantile
  - Quantile indicates the area below the normal distribution curve of error between y_i and f_i.
  - The (P10/P50/P90) quantile represents values with a loss value of less than (10%/50%/90%).