Appendix

Anonymous submission

A. Conditional diffusion for MTS prediction

The process of predicting noise first and then generating target data in diffusion models is as follows.

First of all, by continuously adding Gaussian noise to the ground truth data x_0, we get the pure noise data x_T, that is:

$$q(x_{0:T}|x_0) = \prod_{t=1}^{T} q(x_t|x_{t-1}) \tag{1}$$

where $q(x_t|x_{t-1}) := N(x_t; \sqrt{1-\beta_t}x_{t-1}, \beta_t I)$, β_t is the noise schedule, T and t are diffusion total timesteps and current timestep, respectively. By accumulating the product $\overline{\alpha}_t = \prod_{t=1}^{T} \alpha_t$ (where $\alpha_t = (1-\beta_t)$), we can rewrite equation (1) as follows.

$$q(x_T|x_0) := N(x_T; \sqrt{\overline{\alpha}_t}x_0, (1-\alpha_t)I) \tag{2}$$

The reverse process is to train a neural network to gradually restore the original signal x_0 from x_T, that is:

$$p_\theta(x_0|x_T) = p(x_T) \prod_{t=1}^{T} p_\theta(x_{t-1}|x_t) \tag{3}$$

where $p_\theta(x_{t-1}|x_t) := N(x_{t-1}; \mu_\theta(x_t, t), \sigma_t I)$, $x_T := N(0, I)$, $x_t = \sqrt{\overline{\alpha}_t}x_0 + (1-\alpha_t)\epsilon$ and ϵ denotes the noise added to the input in the forward process. The parameters of μ_θ and σ_t can be computed followed by Ho *et al* (Ho, Jain, and Abbeel 2020).

The reverse process can be trained by solving the following optimization problem:

$$min\, \mathbb{E}_{t,x_0}[||\epsilon - \epsilon_\theta(x_t, t)||^2] \tag{4}$$

where ϵ denotes the noise during the noise addition process (i.e., noise in equation (2)) and $\epsilon_\theta(x_t, t)$ denotes the estimated noise.

If there exists conditional data y in the reverse process of time series generation task, we can run gradient update on x_{t-1} via the following score function (Yuan and Qiao 2024):

$$\begin{gathered} \nabla_{x_{t-1}} log\, p(x_{t-1}|x_t, y) = \\ \nabla_{x_{t-1}} log\, p(x_{t-1}|x_t) + \nabla_{x_{t-1}} log\, p(y|x_{t-1}) \end{gathered} \tag{5}$$

where $log\ p(x_{t-1}|x_t)$ is defined in diffusion models and $log\ p(y|x_{t-1}) \propto -||f(x_{t-1}) - y||_2^2$, f represents the mapping relationship between x_{t-1} and y. Then we can modified reverse diffusion process as follows:

$$\begin{gathered} p_\theta(x_{t-1}|x_t, y) := \\ N(x_{t-1}; \mu_\theta(x_t, t) + s\sigma_t \nabla_{x_{t-1}} log\, p(y|x_{t-1}), \sigma_t I) \end{gathered} \tag{6}$$

where s is a scale parameter controlling the strength of the guidance.

Table 1: DETAILS OF THE FOUR DATASETS

	$Media$	SN	$GAIA$	$AIOps18$
Number of samples	17,257	29,873	8,640	21,180
Number of services	32	26	1	16
Collection particle level	10s	10s	5m	1m
Concept drift	Yes	Yes	Yes	No
Contain RPS	Yes	Yes	No	No

B. Datasets

The details of all datasets that used in this paper is summarized in the Tabel 1.

Specifically, the "Number of samples" represents the volume of collected metric data, while the "Number of services" corresponds to the dimensionality of the services, i.e. the chunks of metrics. The "Collection particle level" denotes the time interval between consecutive metric samplings. The "Concept drift" indicator reflects whether the dataset exhibits metric concept drift. Lastly, "Contain RPS" specifies whether the dataset includes Requests Per Second (RPS) data.

C. The codes of pre-processing and backbone

For the codes of pre-processing and the core codes of this paper, please refer to "https://anonymous.4open.science/r/RCDMMP-2F2B". File_a.py is the the core code of the diffusion model in the paper, File_b.py is the backbone of the diffusion model and File_c.py contains the pre-processing code.

D. The prediction effect of RCDMMP under different prediction step size Δt

We present the prediction effect of RCDMMP under different prediction step size Δt in Table **??**.

As presented in Table **??**, a reduction in the prediction step size does not consistently enhance forecasting performance, which runs counter to conventional expectations. This phenomenon may be attributed to the fact that shorter step sizes potentially neglect underlying periodic structures in the data.

E. The prediction effect of RCDMMP under different observation window lengths w

Furthermore, we present the prediction effect of RCDMMP under different observation window lengths w in Table **??**.

As shown in Table **??**, RCDMMP achieves its best prediction performance across all datasets when the window size w is set to 30. This result is intuitive: a shorter window provides insufficient information for RCDMMP to learn meaningful predictive patterns, while a longer window may incorporate outdated information due to concept drift in the metrics. The inclusion of such obsolete data introduces noise, which can degrade the model's performance.

F. Comparison of prediction effects

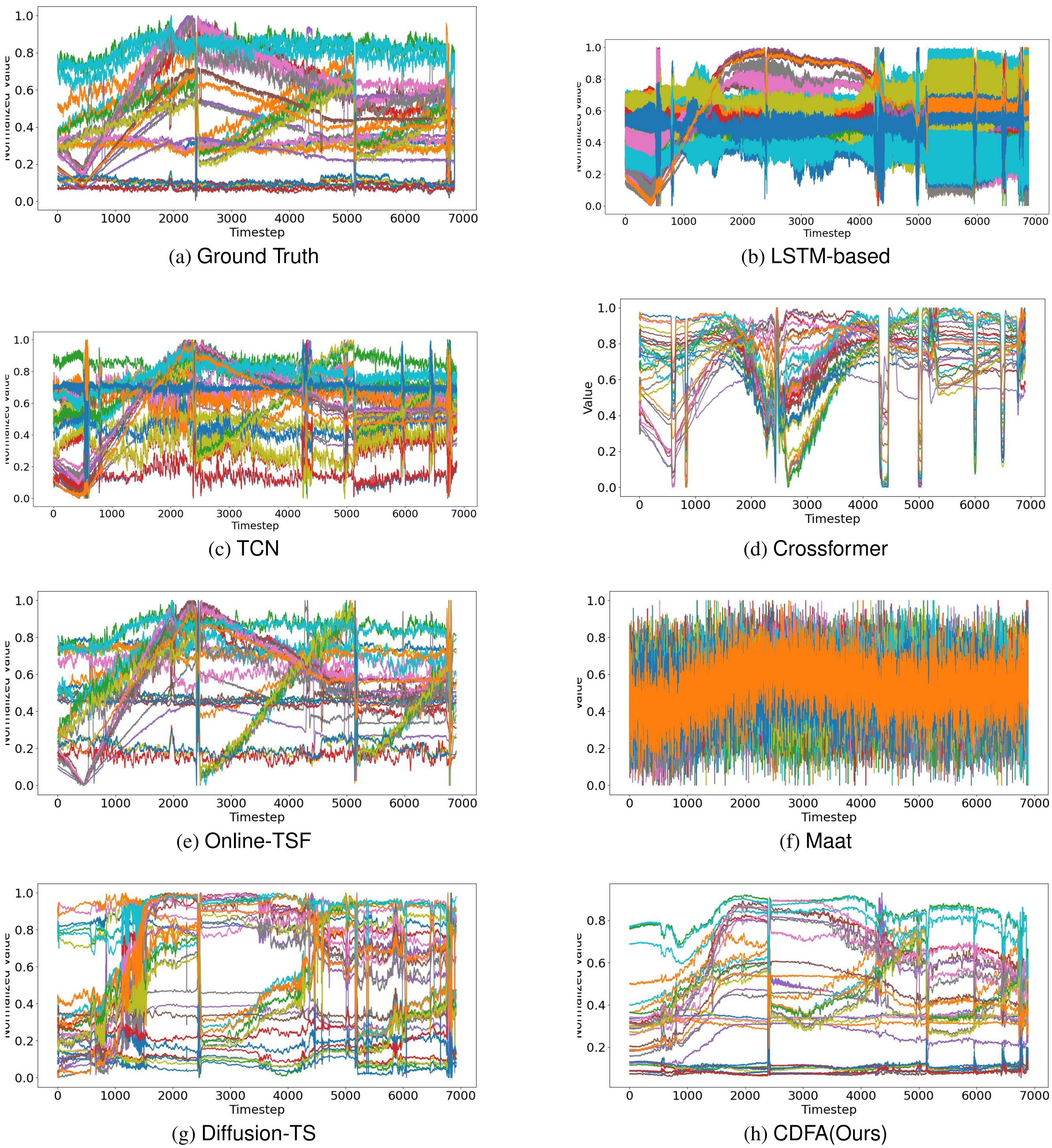
To provide a clearer comparison of the metric prediction performance between RCDMMP and the baseline methods, we consolidate the prediction results from all approaches across the test sets of all datasets and present them collectively.

Specifically, the ground truth result of each datasets is shown in Figure **??** with all data been normalized. And the following Figure 1 to 4 shows the prediction effects of each approaches on the four datasets.

References

Ho, J.; Jain, A.; and Abbeel, P. 2020. Denoising diffusion probabilistic models. *Advances in neural information processing systems*, 33: 6840–6851.

Yuan, X.; and Qiao, Y. 2024. Diffusion-ts: Interpretable diffusion for general time series generation. *arXiv preprint arXiv:2403.01742*.



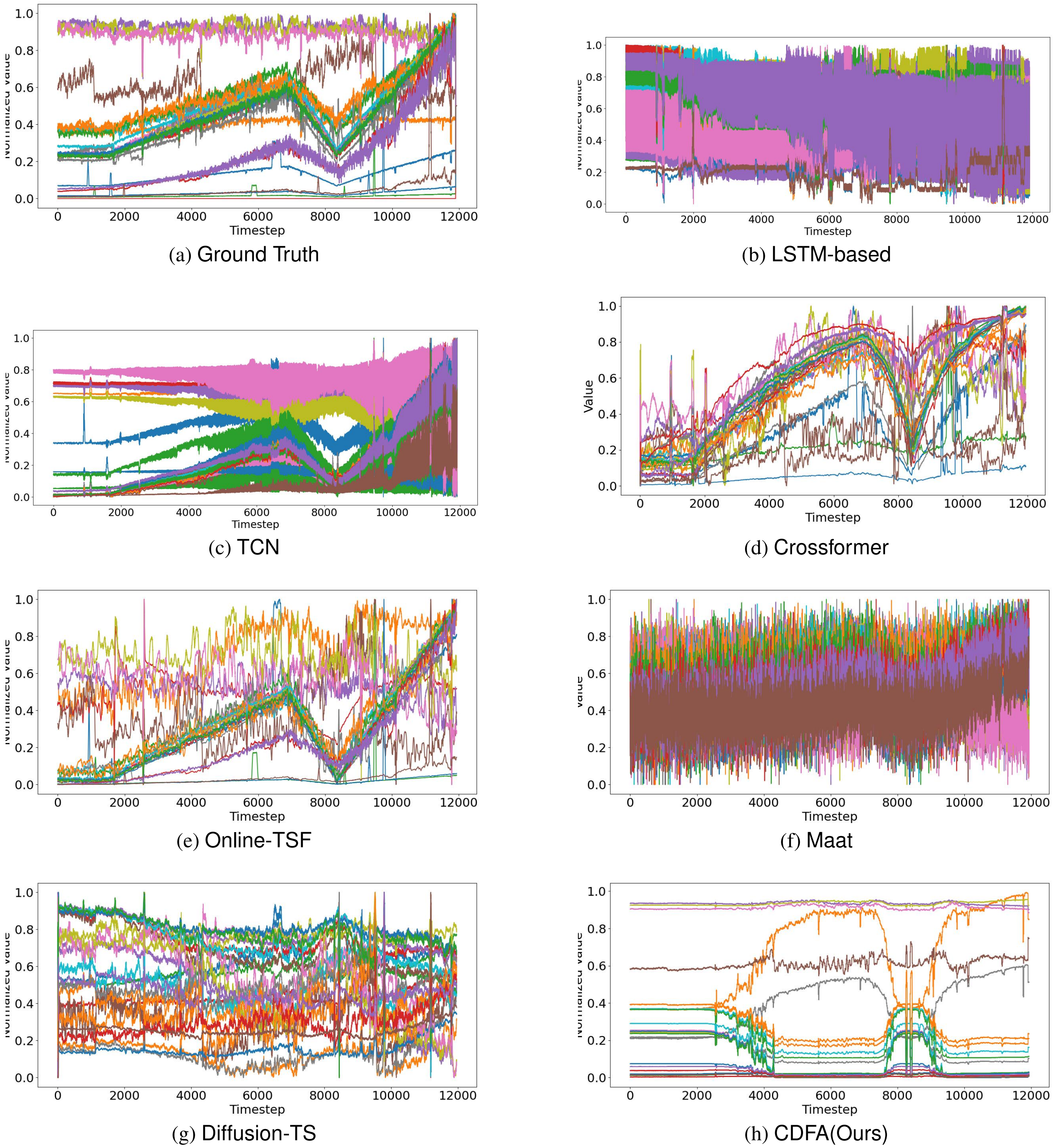
(a) Ground Truth (b) LSTM-based

(c) TCN (d) Crossformer

(e) Online-TSF (f) Maat

(g) Diffusion-TS (h) CDFA(Ours)

Figure 1: The ground truth and prediction results of all methods on Med.



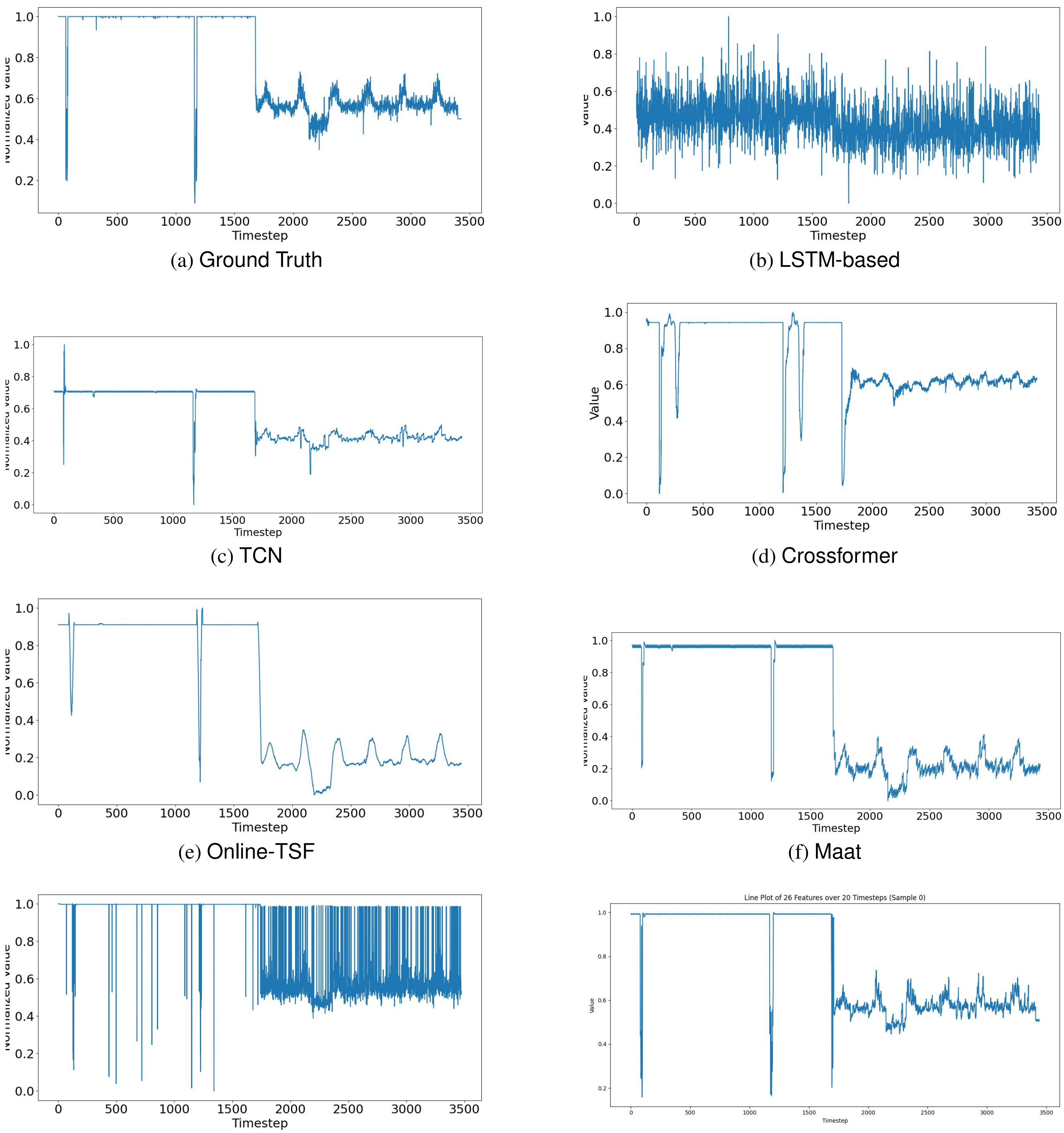
(a) Ground Truth (b) LSTM-based

(c) TCN (d) Crossformer

(e) Online-TSF (f) Maat

(g) Diffusion-TS (h) CDFA(Ours)

Figure 2: The ground truth and prediction results of all methods on SoN.



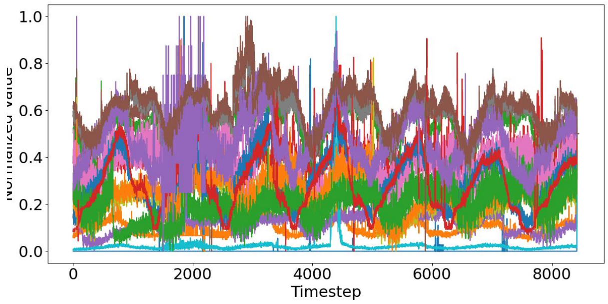
(a) Ground Truth (b) LSTM-based

(c) TCN (d) Crossformer

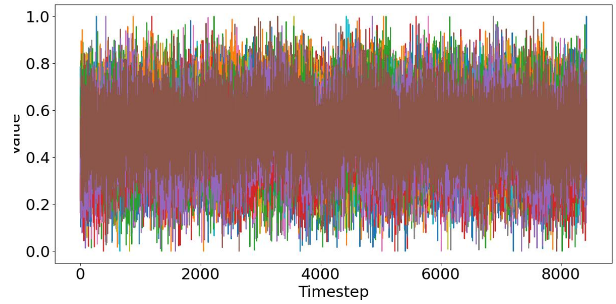
(e) Online-TSF (f) Maat

(g) Diffusion-TS (h) CDFA(Ours)

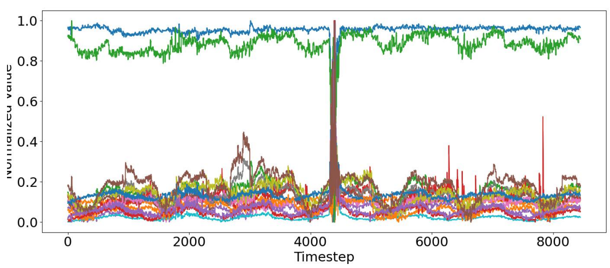
Figure 3: The ground truth and prediction results of all methods on $GAIA$.



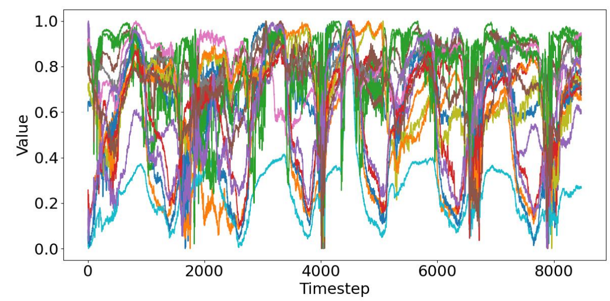
(a) Ground Truth



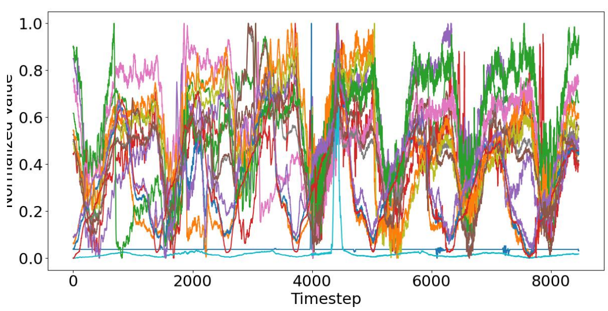
(b) LSTM-based



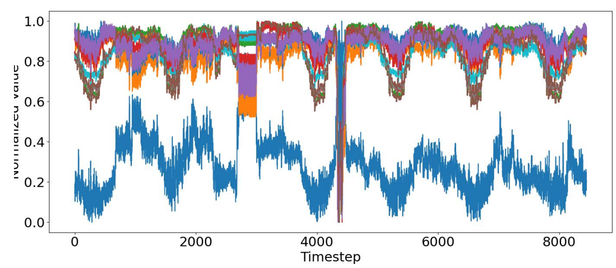
(c) TCN



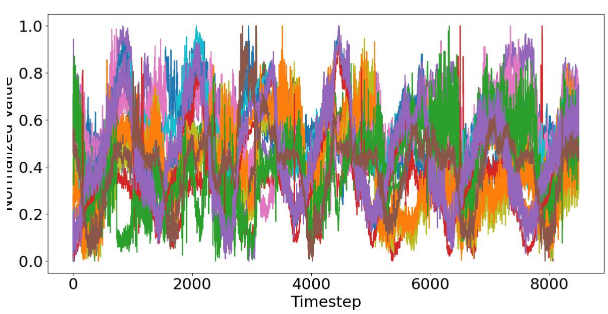
(d) Crossformer



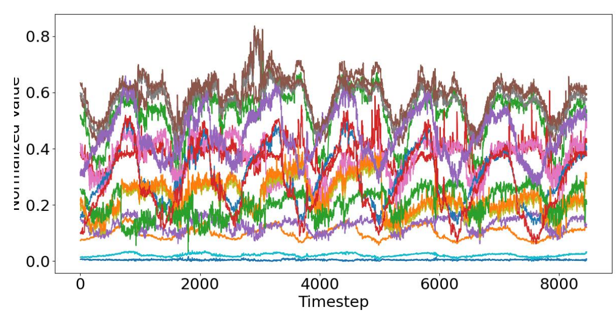
(e) Online-TSF



(f) Maat



(g) Diffusion-TS



(h) CDFA(Ours)

Figure 4: The ground truth and prediction results of all methods on $AIOps18$.