A Training details

We use ten memory items for our MEMTO model, corresponding to the number of clusters in our K-means clustering. We elaborate on our process for deciding the number of clusters in Appendix C.2. To determine anomalies, we set the threshold as the top-$p\%$ of the combined results of the anomaly scores from both the training and validation data, with specified values of p for each dataset outlined in Table 5, following [40]. We set λ in the objective function to 0.01, use Adam optimizer [15] with a learning rate of 5e-5, and employ early stopping with the patience of 10 epochs against the validation loss during training. Our experiments are conducted using the Pytorch framework on four NVIDIA GTX 1080 Ti 12GB GPUs. Furthermore, during the execution of our experiment, we make partial references to the code of [40].

A.1 Hyperparameter settings

Important hyperparameters of MEMTO were determined through grid search, while others were set to commonly used default values based on empirical observations. We performed a grid search to determine the values of each hyperparameter within the following range:

- $\lambda \in$ {1e+0, 5e-1, 1e-1, 5e-2, 1e-2, 5e-3, 1e-3}
- $lr \in$ {1e-4, 3e-4, 5e-4, 1e-5, 3e-5, 5e-5}
- $\tau \in$ {0.1, 0.3, 0.5, 0.7, 0.9}
- $M \in$ {5, 10, 15, 20, 25, 30, 35, 40, 45, 50, 55, 60, 65, 70, 75, 80, 85, 90, 95, 100}

, where lr, τ, and M denote the learning rate, the temperature in the softmax function, and the number of clusters, respectively. Since we set the centroids of clusters as memory items, the number of memory items and that of clusters are the same. We set the optimal hyperparameters as follows: λ as 1e-2, lr as 5e-5, τ as 0.1, and M as 10. All experiments in this paper are conducted using the same hyperparameters regardless of the dataset.

A.2 Dataset

Table 5: Details in five benchmarks. The number of samples in the training, validation, and test sets is represented in the columns labeled 'Train,' 'Valid,' and 'Test,' respectively. The '$p\%$' column indicates the anomaly ratio used in the experiment. The 'Dim' column shows the dimension size of the data for each dataset.

	Train	**Valid**	**Test**	**$p(\%)$**	**Dim**
SMD	566,724	141,681	708,420	0.5	38
MSL	46,653	11,664	73,729	1.0	55
PSM	105,984	26,497	87,841	1.0	26
SMAP	108,146	27,037	427,617	1.0	25
SWaT	396,000	99,000	449,919	0.1	53

Table 5 shows the statistical details of datasets used in experiments. We obtained SWaT by submitting a request through `https://itrust.sutd.edu.sg/itrust-labs_datasets/`.

B Algorithm for MEMTO

Algorithm 2 Proposed Method **MEMTO**

```
**Input** $X^s \in \mathbb{R}^{L \times n}$: input sub-series
**Training params** $f_e$: encoder, $f_d$: decoder, $U_\psi, W_\psi \in \mathbb{R}^{C \times C}$: linear projection matrices
$q^s = f_e\left(X^s\right)$    \\ feed-forward encoder, $q^s \in \mathbb{R}^{L \times C}$
$v^s = softmax\left(m\left(q^s\right)^T\right)$    \\ Gated memory update start, $m \in \mathbb{R}^{M \times C}$, $v^s \in \mathbb{R}^{M \times L}$
$\psi = sigmoid\left(mU_\psi + \left(v^s q^s\right) W_\psi\right)$    \\ $\psi \in \mathbb{R}^{M \times C}$
$m = (1 - \psi) \circ m + \psi \circ \left(v^s q^s\right)$    \\ Gated memory update end
$w^s = softmax\left(q^s\left(m\right)^T\right)$    \\ Query update start, $w^s \in \mathbb{R}^{L \times M}$
$\tilde{q}^s = w^s m$    \\ $\tilde{q}^s \in \mathbb{R}^{L \times C}$
$\hat{q}^s = concat\left(\left[q^s, \tilde{q}^s\right], dim = 1\right)$    \\ Query update end, $\hat{q}^s \in \mathbb{R}^{L \times 2C}$
$\hat{X}^s = f_d\left(\hat{q}^s\right)$    \\ feed -forward decoder, $\hat{X}^s \in \mathbb{R}^{L \times n}$
**return** $\hat{X}^s$    \\ reconstructed sub-series
```

Algorithm 2 provides an overall mechanism for our model. It demonstrates the matrix operation version of the forward process when a single input sub-series X^s is fed to MEMTO.

C Additional experiments

Table 6: The ablation results (F1-score) in anomaly criterion and objective function. L_{rec} and L_{entr} signify Reconstruction Loss and Entropy Loss, respectively.

Loss	**Anomaly Criterion**		**F1-score**					
	ISD	**LSD**	**SMD**	**MSL**	**PSM**	**SMAP**	**SWaT**	**avg.**
L_{rec}	✓	×	79.63	86.23	82.15	71.18	31.29	70.09
	×	✓	69.73	72.63	93.07	67.69	82.50	77.12
	✓	✓	93.19	92.66	98.05	96.48	93.34	94.74
L_{entr}	✓	×	75.71	88.39	87.47	69.28	79.28	80.02
	×	✓	12.53	84.34	76.49	68.17	83.52	65.01
	✓	✓	88.43	93.40	97.97	96.22	92.77	93.75
$L_{rec} + \lambda L_{entr}$	✓	×	77.54	87.22	79.25	70.99	31.17	69.23
	×	✓	72.78	80.33	80.15	67.55	0.00	60.16
	✓	✓	**93.54**	**94.36**	**98.34**	**96.61**	**95.83**	**95.73**

C.1 Objective function and anomaly criterion

In this experiment, we investigate the impact of loss terms, specifically the reconstruction loss L_{rec} and the entropy loss L_{entr}, on the performance of our proposed framework, MEMTO. We remove one of the two terms one by one from the objective function and evaluate the resulting performance. Table 6 demonstrates the significance of incorporating both L_{rec} and L_{entr} terms in the objective function. Applying bi-dimensional deviation-based criterion to the MEMTO variants that only use L_{rec} or L_{entr} as the loss function shows competitive performance compared to ours in terms of average F1-score. This demonstrates the robustness of MEMTO to loss terms. Additionally, both cases show a significant performance drop when using only ISD or LSD as the anomaly criterion, emphasizing the importance of combining ISD and LSD for achieving optimal performance.

C.2 Number of memory items

Figure 4 illustrates the relationship between MEMTO's performance and the number of memory items used. Results reveal that MEMTO's performance is robust to the number of memory items, as

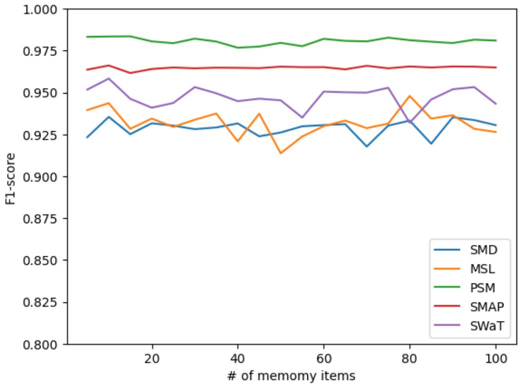

Figure 4: Number of memory items

the performance variance across datasets is small. Nevertheless, increasing the number of memory items raises the number of clusters needed for K-means clustering, thereby increasing computational complexity. Hence, we designate ten memory items as the default value after weighing performance and computational complexity.

Our study highlights the effectiveness of employing a restricted number of memory items to extract prototypical features of normal patterns in time series data. Unlike computer vision, which may require thousands of memory items [8], we demonstrate that only ten memory items were necessary for this task in the time series domain.

C.3 Number of decoder layers

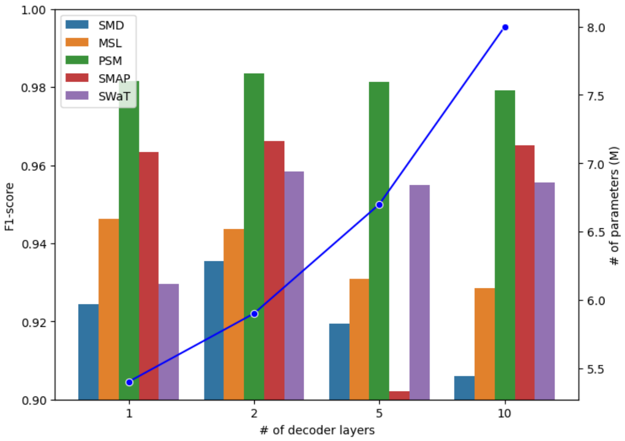

Figure 5: F1-score and number of parameters, according to the number of decoder layers. The right y-axis represents the values of the blue line graph in million units, while the left y-axis represents the values of the bar graph.

Figure 5 provides the performance of MEMTO under different numbers of decoder layers. As shown in Figure 5, a decoder that is too shallow (e.g., a decoder with a single layer) performs worse because it lacks sufficient capacity to reconstruct the input data accurately. On the other hand, if the decoder is too large (e.g., decoder with ten layers), it can become overly expressive and reconstruct even anomalies regardless of the encoding ability of the encoder. Therefore, it can lead to an over-generalization problem, which can ultimately decrease the performance of anomaly detection by reconstructing anomalies too accurately. Furthermore, a larger decoder layer with more parameters can increase computational and memory costs. We empirically find that considering the balance between performance and resource cost, a decoder with two layers is most suitable for anomaly detection tasks presented in our paper.

D Additional details for discussion

D.1 LSD values

Table 7: The mean LSD values corresponding to test data.

	SMD		**MSL**		**PSM**		**SMAP**		**SWaT**	
	Normal	Abnormal	Normal	Abnormal	Normal	Abnormal	Normal	Abnormal	Normal	Abnormal
MemAE	814.7836	842.2023	622.5195	640.4954	766.2473	782.1895	710.4929	706.3115	795.7227	770.9069
MNAD	259.3633	258.0175	791.6371	788.2654	292.3340	293.4836	301.3480	301.2153	303.1933	310.9818
Ours	297.5692	330.1162	249.8632	263.4532	340.7552	363.7520	237.0070	234.7110	450.0926	721.3093

Table 7 shows mean LSD values of normal and abnormal samples across various domains of datasets while using different memory module mechanisms. In most datasets, our proposed Gated memory module consistently exhibits a lower mean LSD value for normal samples than for abnormal samples. Furthermore, the relative difference between these values is more significant than other memory module mechanisms. These results demonstrate the efficacy of our memory module mechanism in capturing prototypical features of normal patterns in data.

D.2 Anomaly score

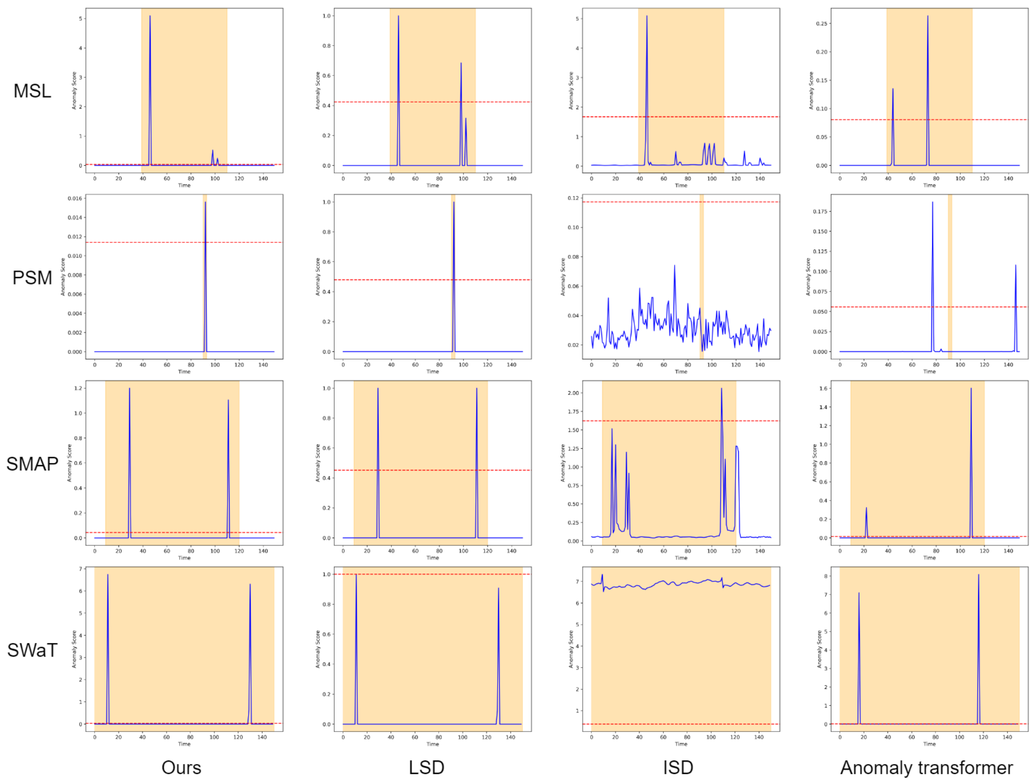

Figure 6: Visualization of anomaly scores for MSL, PSM, SMAP, and SWaT datasets.

Figure 6 visually represents the anomaly scores for benchmark datasets not discussed in Section 4.4. We randomly sampled data of length 150 from MSL, PSM, SMAP, and SWaT test datasets and plotted the anomaly scores for each segment. Compared to other baselines, our proposed method consistently detects anomalies precisely with a low false positive rate from the perspective of the point adjustment method.