7 APPENDIX

7.1 Dataset Description

We use five real-world datasets and one synthetic dataset for comprehensive evaluations:

- Synthetic [36]: It is a synthetic dataset, generated by the MTS generator TSAGen [37].
- MoCap [25]: Derived from the CMU motion capture repository. In this dataset, every motion is represented as a sequence of hundreds of frames.
- ActRecTut [7]: This dataset involves two participants performing hand movements with height gestures in daily life and 3D gestures while playing tennis.
- PAMAP2 [28]: Covering both basic (e.g., walking, sitting) and composite human activities (e.g., soccer), this dataset features data from eight individuals.
- UscHad [43]: This dataset encompasses 12 distinct human activities such as jumping and running, recorded for 14 individuals.
- UcrSeg [18]: This dataset encompasses diverse sources, including medical fields, insect studies, robotics, and power demand data.

A summary of the statistics is provided in Table 3. Specifically, the varying range denoted as $x - y$ for the number of states, i.e., #(State), implies that an individual time series within the MTS can encompass as few as x states and as many as y states. The same applies to the length and state duration.

Table 3: Statistics of the datasets.

Dataset	#(MTS)	#(State)	#(Variate)	Length (k)	State Duration (k)*
Synthetic	100	5	4	9.3-23.7	0.1-3.9
MoCap	9	5-8	4	4.6-10.6	0.4-2.0
ActRecTut	2	6	10	31.4-32.6	0.02-5.1
PAMAP2	10	11	9	253-408	2.0-40.3
UscHad	70	12	6	25.4-56.3	0.6-13.5
UcrSeg	32	2-3	1	2-40	1-25

*The state duration is the range of continuous length of a state based on ground truth.

7.2 Implementation Details

Experiments were conducted on a server with an NVIDIA Quadro RTX 8000 GPU and an Intel Xeon Gold 5215 CPU (2.50GHz). For FFTCompress in Section 3.1.1, the frequency bandwidth $\mathtt{Q}$ is set to 33 (see Equation (4) and Equation (5)). For DDEM in Section 3.1.2, κ in Equation (6) is set to 5, the dimension of the intermediate embeddings $\boldsymbol{h}^{\mathrm{T}}$ and $\boldsymbol{h}^{\mathrm{S}}$ (see Equation (7)), denoted as $\mathtt{C}$, is set to 80 by default, the random convolution kernel for Conv1D in Equation (7) is set to 3, and the dimension of the final embedding $\boldsymbol{z}$ in Equation (7), denoted as $\mathtt{D}$ is set to 4. The FNCCLearning method used $\mathtt{U} = 20$ groups of windows (each with $\mathtt{V} = 4$ neighboring windows) and a fraction threshold λ of 0.5. Sliding window sizes were 128, 256, or 512, dataset-dependent, with step size $\mathtt{B} = 50$. Adam optimizer [22] was used with a learning rate of 0.003 for 20 epochs. Lastly, AdaTD was configured with scaling factors $\delta_i = 0.08$ and $\delta_r = 0.1$ and initiated the threshold τ at a value of 1. The sensitivity of various key parameters, including $\mathtt{U}$, $\mathtt{V}$, $\mathtt{Q}$, $\mathtt{D}$, $\mathtt{C}$, δ_i, and δ_r are reported later in this appendix.

The MCU deployment uses an STM32H747 device [2] with a 480 MHz Arm Cortex-M7 core, 2 MB Flash memory, and 1 MB RAM, as presented in Fig. 8. The E2USD model was converted to ONNX format and translated to C code with X-CUBE-AI [3]. The C code was compiled using an ARM-specific version of GCC to create an executable binary.

Fig. 8: The STM32H747 device for model deployment.

7.3 Impact Assessment of the Energy-based Frequency Compressor

To verify the effectiveness of the Energy-based Frequency Compressor (EFC), we conduct FFTCompress on the UcrSeg dataset using various bandwidth values, denoted as $\mathtt{Q} = [120, 60, 40]$. Referring to Fig. 9, the top row showcases the original and reconstructed waveforms within the native time domain. The middle row displays their corresponding amplitude spectra, while the bottom row exhibits the compressed waveforms. The original data is represented in blue, the reconstructed versions in orange, and the compressed versions in green.

Upon reverting the filtered frequency components to the original time domain, we observe *minimal distortion*. Specifically, the Mean Absolute Percentage Error (MAPE) is less than 5%, even though we retain only a sixth of the original frequency domain representation.

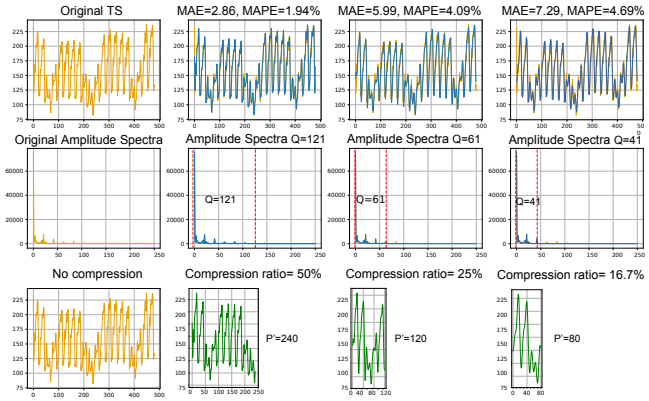
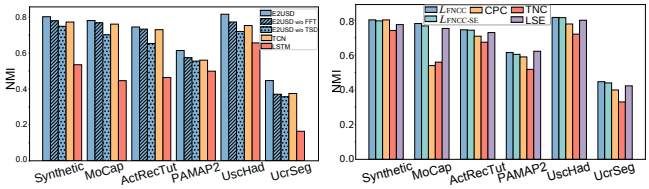


Fig. 9: Impact assessment of the Energy-based Frequency Compressor on the performance of FFTCompress.

7.4 Additional NMI Results for Component Study

In Fig. 10, we present the NMI comparisons for both encoders and losses. We note that the trends in NMIs are basically consistent with the corresponding ARIs shown in Fig. 6. This observation further substantiates the effectiveness of our proposed encoder and loss components.



(a) Encoder (b) Loss

Fig. 10: NMI comparison.

7.5 Parameter Sensitivity Study

We conduct a comprehensive parameter analysis using the ActRecTut dataset, focusing on assessing how these key parameters affect the ARI and NMI.

7.5.1 Impact of U *and* V *in Negative Sampling.* These two parameters jointly contribute to the computation of $\mathcal{L}_{\text{FNCC}}$. More specifically, U designates the number of distinct window groups, whereas V defines the number of consecutive windows within each group. As shown in Fig. 11, altering V does not consistently influence ARI. This unexpected result could be attributed to larger V values capturing broader temporal scopes, thereby introducing false positives due to state transitions. This implies that while increasing V appears to be beneficial for capturing more data, it may inadvertently degrade performance. Conversely, ARI remains stable across a range of U values, highlighting the robustness of E2USD.

7.5.2 Impact of Frequency Bandwidth Q *in* FFTCOMPRESS. This parameter Q serves as the size of the frequency bandwidth of Energy-based Frequency Compression (EFC) on the FFTCOMPRESS. It has a direct bearing on the FFTCOMPRESS's compression rate. Fig. 12 exhibits two key trends. Lower Q values compromise ARI and NMI due to aggressive data compression, causing the loss of essential information. On the other hand, elevating Q leads to performance plateaus or minor reductions, likely because of the introduction of noise.

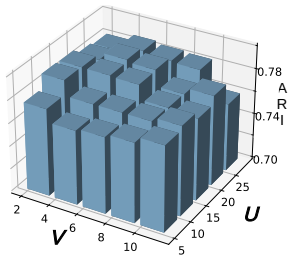


Fig. 11: Impact of U and V.

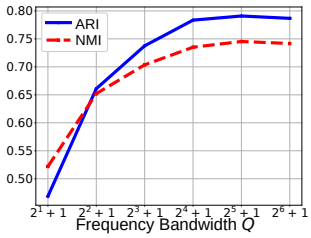


Fig. 12: Impact of Q.

7.5.3 Impact of Intermediate Embedding Size C *in* DDEM. As illustrated in Fig. 13, enlarging the latent space dimensionality C generally boosts both ARI and NMI, peaking at C = 80. Further increases in C result in diminishing returns and even minor performance setbacks. Thus, by default, we set C to 80 for all the experiments.

7.5.4 Impact of Final Embedding Size D *in* DDEM. As demonstrated in Fig. 14, increasing the embedding size D typically enhances ARI and NMI metrics, with a peak performance observed at D = 4. Beyond this value, the benefits of enlarging D decrease, and there may even be slight performance deterioration. By default, we set D = 4 in E2USD.

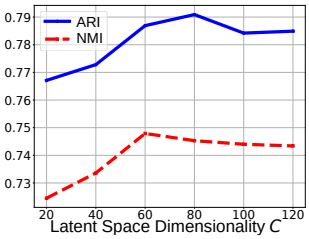


Fig. 13: Impact of C.

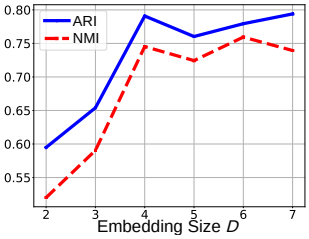
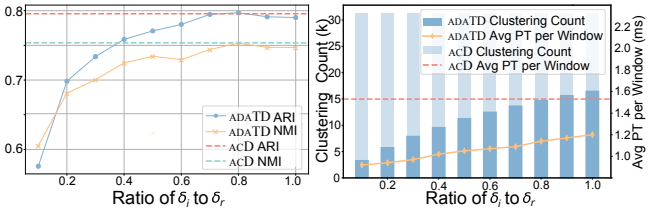


Fig. 14: Impact of D.

7.5.5 Impact of δ_i and δ_r in ADATD. In E2USD, the adaptability of ADATD stems from its ability to adjust the threshold τ based on the model's response, which is directly influenced by the $\frac{\delta_i}{\delta_r}$ ratio. For our evaluation, we set $\delta_r = 0.1$ and adjust the $\frac{\delta_i}{\delta_r}$ ratio within a range of 0.1 to 1.

As shown in Fig. 15, the effectiveness of ADATD improves incrementally with an increase in the $\frac{\delta_i}{\delta_r}$ ratio. Remarkably, it nears parity with the conventional "Always Clustering Detection" (ACD) approach when $\frac{\delta_i}{\delta_r} = 0.8$. This is achieved while executing notably fewer clustering operations and maintaining a reduced average processing time per window.



(a) Effectiveness (b) Efficiency

Fig. 15: Impact of the ratio δ_i to δ_r.