A Two-Stage Framework for Time-series Clustering with Encoder-agnostic Compatibility

Hyuntae Kim, Eun Seo Lee, Hyun-Chul Kang, and Ji-Yeon Son
ICT-enabled Intelligent Manufacturing Research Section
Electronics and Telecommunications Research Institute
Daejeon, Korea
{kht, eslee, kauni, jyson}@etri.re.kr

***Abstract*—Unsupervised clustering of time-series data is crucial in domains where labeled data is unavailable and discovering meaningful subgroups within sequential data is essential for decision-making. However, most existing approaches are tightly bound to a specific representation learning paradigm, limiting their adaptability to varying data characteristics and architectures. In this paper, we propose a modular and encoder-agnostic clustering framework that can be seamlessly integrated with diverse self-supervised representation learning backbones. Our method follows a two-stage pipeline: we first pretrain a temporal encoder using self-supervision to learn expressive representations, then fine-tune the encoder jointly with a clustering objective to shape a cluster-aware latent space. We further enhance training stability by employing a batch-wise centroid update mechanism compatible with mini-batch iteration. Experimental results on multiple real-world time-series datasets show that our method consistently outperforms a baseline that relies solely on pretrained embeddings without incorporating clustering objective. We validate the effectiveness of our modular clustering framework through t-SNE-based visual analysis and rigorous quantitative evaluation.**

***Index Terms*—clustering, time-series data**

I. Introduction

Clustering time-series data in an unsupervised manner is a critical task across many real-world domains, where explicit class labels are unavailable or prohibitively expensive to obtain. Rather than relying on predefined categories or supervised guidance, clustering enables the discovery of intrinsic groupings in data, which is particularly valuable in domains where the underlying pattern taxonomy is unknown. For time-series data, which are inherently temporal and often high-dimensional, this requires robust representation learning combined with adaptive clustering strategies.

In manufacturing, multivariate time-series data are abundantly generated from industrial equipment such as sensors, programmable logic controllers (PLCs), and power meters. These data reflect dynamic changes in machine operations and process states, yet are rarely annotated. Unsupervised clustering provides a practical solution for identifying distinct operational modes and even KPI-driven analysis such as OEE computation, without the need for labeled ground truth [1]. In healthcare, continuous time-series signals such as ECG, respiration curves, and glucose monitoring data are widely available for patient monitoring. However, the lack of well-defined categories and the complexity of patient-specific temporal patterns hinder supervised modeling. Clustering can help group patients exhibiting similar physiological patterns and uncover subtypes of diseases, empowering healthcare practitioners to make informed clinical decisions [2, 3]. Such numerous applications across various domains have led to a growing demand for clustering methods capable of applying to diverse data types spanning different domains while effectively capturing their heterogeneous temporal patterns.

Despite its importance, time-series clustering remains a challenging task due to the intrinsic diversity of temporal data. Most existing clustering approaches are tightly optimized for a single representation learning encoder, which inherently limits their applicability to specific types of time-series data. When temporal characteristics, such as sampling frequency, stationarity, or pattern complexity, differ from those seen during model design, the encoder often fails to produce meaningful representations, leading to significant degradation in clustering performance. This phenomenon reflects a fundamental limitation in machine learning: *there is no one-fits-all method*. Therefore, a more flexible and modular design is needed in which the clustering functionality remains stable regardless of the underlying encoder architecture. To address this challenge, we propose a 'plug-and-play' clustering framework that is decoupled from any specific encoder. Our method enables adaptive integration with diverse temporal backbones, allowing a representation-compatible and encoder-agnostic clustering framework that can adapt across diverse data domains. Throughout experiments on real-world time-series datasets, we validate the effectiveness of our proposed method by demonstrating superior clustering performance compared to a baseline that relies solely on pretrained embeddings without clustering-specific learning. Furthermore, qualitative analyses using latent space visualizations demonstrate that our method successfully captures meaningful temporal structures and preserves cluster separability across diverse time-series patterns.

II. Related works

The proposed method adopts a two-stage framework that consists of: (i) time-series representation learning module to extract semantically meaningful embeddings and (ii) fine-tuning a clustering module on top of the learned embeddings to induce cluster-friendly representations and obtain explicit

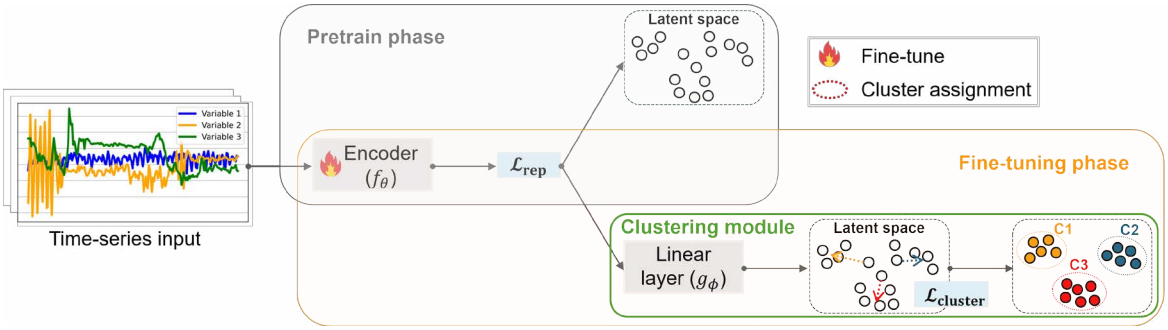

Fig. 1. Overall network architecture of our method.

cluster assignments. We review relevant literature along these two axes.

A. Time-series Representation Learning

Recent advances in computer vision and natural language processing have highlighted the effectiveness of large-scale representation learning and transfer learning frameworks. Motivated by this success, representation learning for time-series data has gained increasing attention. These methods aim to learn temporally aware embeddings that capture local and global structures inherent in sequential signals. Among them, TS2Vec [4] stands out as a state-of-the-art self-supervised learning framework for time series, which learns contextualized representations by maximizing the similarity between representations of overlapping subsequences across multiple temporal resolutions using a hierarchical contrastive loss. Although TS2Vec yields powerful embeddings that improve downstream tasks such as classification or forecasting, it does not explicitly promote cluster-friendly properties in the learned space. Moreover, the representation remains a black-box, limiting interpretability and direct cluster assignment in an unsupervised setting. To address this issue, our method integrates both representation learning and clustering objective during fine-tuning. By jointly optimizing a general-purpose representation loss and a clustering-specific objective, we induce cluster-friendly latent representations while preserving the semantic structure encoded in the pretrained embeddings. This prevents the encoder from collapsing into trivial solutions, thereby enabling stable training and yielding more accurate cluster assignments.

B. Deep Clustering Method

Deep clustering method aims to jointly optimize discriminative feature extractor and cluster assignments through a unified training objective. Notable examples are the deep clustering network (DCN) [5] and its variant [6], which propose a joint learning framework in which latent features are shaped to be compatible with K-means clustering. This is typically achieved by integrating a clustering loss into an autoencoder or reconstruction-based architecture, encouraging the intermediate features to form compact clusters. Although DCNs have shown effectiveness in image and tabular domains, typically relying on reconstruction-based architectures, their applicability to time-series data remains limited. These methods are often designed for static data and do not adequately address the temporal dependencies and dynamic variations inherent in sequential signals. As a result, the interaction between time-series-specific backbones and clustering modules has not been thoroughly explored or validated in prior literature. In this work, we bridge this gap by demonstrating that clustering-aware objective can be stably integrated with pretrained temporal encoders through a plug-and-play design. Our method successfully couples a clustering module with a time-series backbone, enabling robust cluster discovery without compromising the quality of the temporal representation. This modular integration provides a practical pathway toward scalable, encoder-agnostic time-series clustering.

III. Method

A. Problem Formulation

Let $\mathcal{D}_{\text{tr}} = \{x_i\}_{i=1}^{N_{\text{tr}}}$ be a training dataset consisting of multivariate time-series samples, where each $x_i \in \mathbb{R}^{L\times D}$ represents a time-series of length L with D variables. The test dataset is denoted as $\mathcal{D}_{\text{te}} = \{(x_i, y_i)\}_{i=1}^{N_{\text{te}}}$, where $y_i \in \{0, \dots, C-1\}$ is the ground-truth class label used only for evaluation.

We consider an fully unsupervised setting where no label information is available during training. The goal is to learn a clustering assignment function, i.e., $P : \mathbb{R}^{L\times D} \to \{0, \dots, K-1\}$, which assigns unseen test input x^* to one of K clusters representing distinct temporal patterns. For the proposed unsupervised clustering framework, we introduce three components – an encoder, a projection head, and a clustering module – as shown in Figure 1. We define each component as follows:

- *Encoder* $f_\theta : x \mapsto z \in \mathbb{R}^Z$ maps the input sequence x into a latent representation z.
- *Projection head* $g_\phi : z \mapsto h \in \mathbb{R}^H$ is a linear layer that projects the latent representation z into a clustering-compatible space.
- *Clustering module* learns K centroids $M = [\mu_1, \dots, \mu_K] \in \mathbb{R}^{H\times K}$ and minimizes the assignment loss to form compact clusters in the latent space.

B. Pretraining via Self-supervision

Our method follows a two-stage training scheme, where the encoder network f_θ is first pretrained using a self-supervised

learning objective before clustering optimization is applied. This pretraining phase initializes the encoder parameters θ in a way that preserves temporal and structural patterns embedded in the raw time-series data, thereby enhancing its ability to capture meaningful variations among latent representations. The encoder architecture f_θ can be flexibly selected depending on the data characteristics, including convolutional, recurrent, or attention-based models. The pretraining objective, denoted by $\mathcal{L}_{\text{rep},i} = \ell(f_\theta(x_i))$, can also be instantiated using various self-supervised frameworks such as TS2Vec or reconstruction-based losses. These objectives are designed to learn representations that reflect both temporal patterns and high-level global structures across the input sequences.

C. Fine-tuning with Clustering Objective

Following the self-supervised pretraining phase, we fine-tune the encoder to promote clustering-aware representations while preserving the temporal semantics learned previously. To this end, we define a clustering loss that encourages the projected representation h_i to be close to one of K latent cluster centroids. Let $M = [\mu_1, \ldots, \mu_K] \in \mathbb{R}^{H\times K}$ denote the matrix of cluster centroids, and $s_i \in \{0,1\}^K$ be a one-hot assignment vector indicating the cluster to which h_i is assigned, satisfying $\sum_{k=1}^{K} s_{i,k} = 1$. The clustering loss is defined as:

$$\mathcal{L}_{\text{cluster},i} = \sum_{i=1}^{N_{\text{tr}}} \|h_i - M s_i\|_2^2 . \tag{1}$$

This objective encourages each sample to be mapped close to its assigned centroid in the latent space, effectively shaping the encoder output into a cluster-friendly representation space. To avoid degenerate solutions such as all samples collapsing to a single cluster or the encoder producing trivial outputs, we retain the self-supervised representation loss $\mathcal{L}_{\text{rep}}$ during fine-tuning. This ensures that the encoder continues to preserve meaningful temporal structure while being refined for clustering. Thus, we jointly train the three components of our unsupervised clustering framework – f_θ, g_ϕ, and K centroids M – based on the following objective:

$$\underset{\theta,\phi,M}{\text{minimize}} \ \frac{1}{N_{\text{tr}}} \sum_{i=1}^{N_{\text{tr}}} \mathcal{L}_{\text{rep},i} + \lambda \mathcal{L}_{\text{cluster},i}. \tag{2}$$

where $\lambda > 0$ is a balancing hyperparameter that controls the trade-off between preserving the pretrained representation and promoting clustering structure.

D. Batch-wise Update of Cluster Centroid

Following the K-means-friendly clustering approach proposed in the Deep Clustering Network (DCN) [5], we update both cluster assignments and cluster centroids $M = [\mu_1, \ldots, \mu_K] \in \mathbb{R}^{H\times K}$ during training.

Cluster assignment update. For each projected feature $h_i = g_\phi(f_\theta(x_i))$, the assignment vector $s_i \in \{0,1\}^K$ is determined by assigning h_i to the centroid with the smallest Euclidean distance among the K centroids:

$$s_{i,k} = \begin{cases} 1 & \text{if } k = \arg\min_j \|h_i - \mu_j\|_2^2 , \\ 0 & \text{otherwise.} \end{cases} \tag{3}$$

This hard assignment corresponds to assigning h_i to the nearest centroid in the latent space.

Cluster centroid update. Once the assignment s_i is computed, the centroid corresponding to the assigned cluster is updated immediately using an online exponential moving average (EMA) strategy:

$$\mu_k^{(t+1)} \leftarrow \mu_k^{(t)} + \eta \left(h_i - \mu_k^{(t)}\right) s_{i,k}, \tag{4}$$

where $\eta \in [0,1]$ is a centroid-specific learning rate, and $s_{i,k}$ ensures that only the centroid associated with the assigned cluster is updated for the current sample h_i. This batch-wise and online update mechanism provides several practical advantages. It enables gradual convergence of cluster structures without requiring full-dataset passes and aligns naturally with mini-batch stochastic gradient descent (SGD). In addition, since centroids are continuously refined using only current batch samples, the model can stably adapt to evolving data distributions and maintain consistent cluster structure throughout training.

IV. Experiments

TABLE I
CLUSTERING PERFORMANCE COMPARISON ON THE FOUR DATASETS. THE BEST METHOD IS HIGHLIGHTED IN BOLD.

Dataset	NMI		RI	
	TS2Vec (W/O fine-tuning)	**TS2Vec-Ours (W/ fine-tuning)**	TS2Vec (W/O fine-tuning)	**TS2Vec-Ours (W/ fine-tuning)**
Natops	0.318	**0.721**	0.766	**0.875**
Epilepsy	0.316	**0.478**	0.733	**0.767**
UMD	0.475	**0.532**	0.716	**0.725**
BasicMotion	0.940	**1.000**	0.976	**1.000**

A. Experimental Setup

We evaluate our proposed clustering framework on a set of multivariate time-series benchmarks selected from the UCR Time Series Classification Archive [7]. These datasets are originally designed for classification, with each sample annotated by a ground-truth class label. This makes them well-suited for evaluating clustering performance in an unsupervised setting. During training, only the raw time-series inputs are used without any label supervision. For quantitative assessment, we report two widely used clustering metrics: normalized mutual information (NMI) [8] and Rand index (RI) [9], which measure the alignment between predicted clusters and actual class distributions.

B. Qualitative Analysis via t-SNE Visualization

To qualitatively assess the learned representations and clustering structures, we visualize the latent embeddings using t-SNE projection. Fig. 2 shows scatter plots for four datasets—BasicMotion, Epilepsy, Natops, and UMD—each

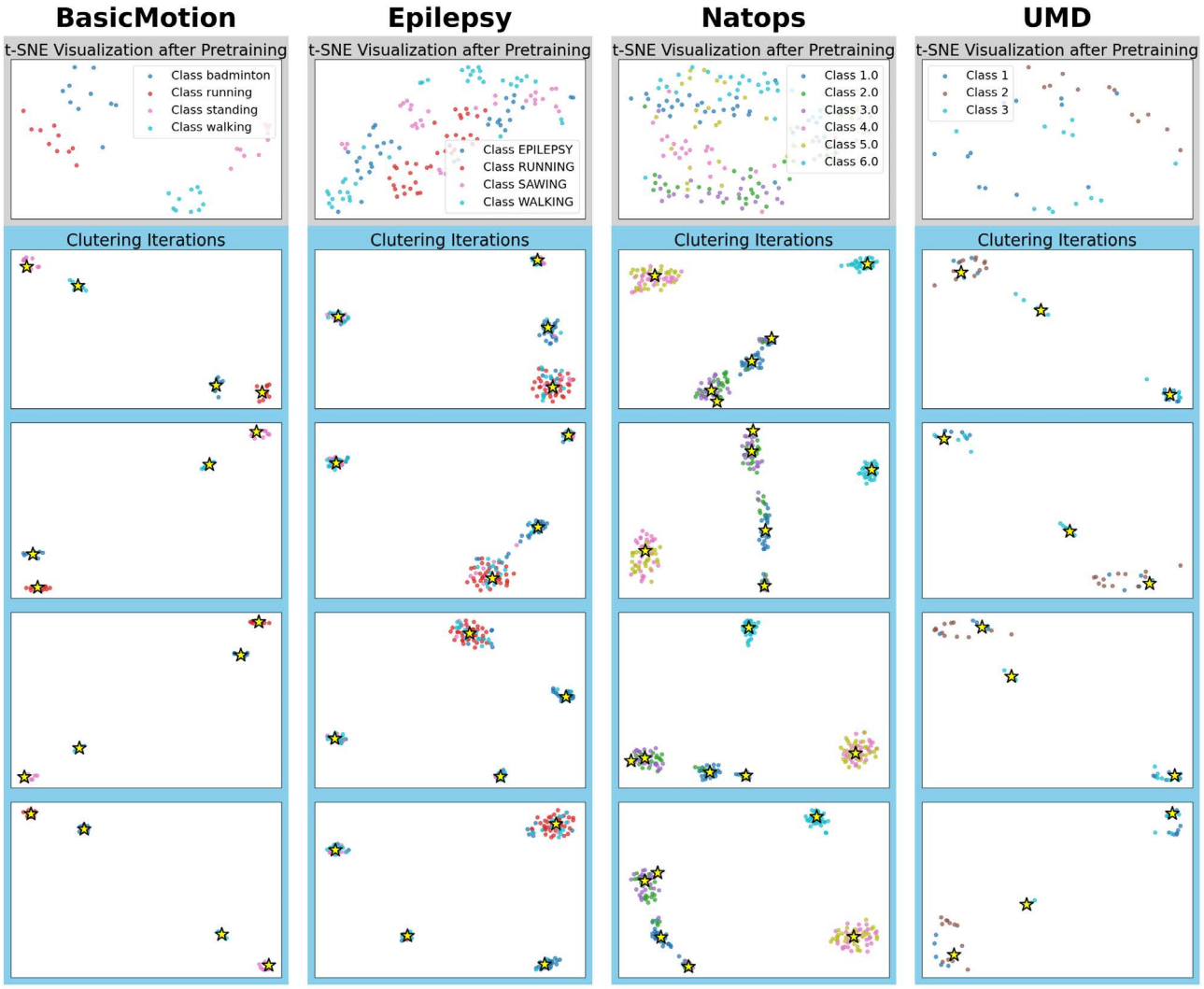

Fig. 2. t-SNE visualizations of latent representations across clustering iterations for four datasets (columns). The top row shows pretrained embeddings before clustering, while subsequent rows illustrate representation changes during clustering fine-tuning. Colors indicate ground-truth class labels; yellow stars denote cluster centroids.

with ground-truth class labels color-coded. The top row for each dataset corresponds to the state of the representation space immediately after self-supervised pretraining, before fine-tuning is applied. Although some class separability is observed, the space is not yet cluster-friendly, and no cluster assignments are available at this stage. From the second row onward, we visualize the latent space after several clustering iterations. As training progresses, the embedding space becomes increasingly structured: cluster boundaries sharpen, inter-cluster distances increase, and class-specific groups become more compact. The yellow star markers represent the cluster centroids, which are dynamically updated through our batch-wise strategy. Their movement over iterations reflects the evolving cluster structure guided by our joint optimization of representation and clustering objectives. This progressive organization of the latent space demonstrates the model's ability to form meaningful clusters that align well with the underlying class structure, despite being trained in a completely unsupervised manner.

C. Quantitative Evaluation

We compare the proposed method against a baseline that directly applies K-means algorithm to pretrained TS2Vec embeddings without any clustering-specific fine-tuning. As shown in Table I, our approach, i.e., TS2Vec-Ours (W/ fine-tuning), consistently outperforms the baseline, i.e., TS2Vec (W/O fine-tuning), across all four datasets in terms of both normalized mutual information (NMI) and Rand index (RI). Notably, the BasicMotion dataset achieves nearly perfect scores in both settings, but our method still pushes performance from 0.940 to 1.000 in NMI and from 0.976 to 1.000 in RI, indicating robust alignment between clustering assignments and ground-truth labels. These results confirm that the proposed clustering framework, which jointly optimizes both representa-

tion learning and clustering objectives, yields more cluster-discriminative embeddings compared to simple pretraining.

V. Conclusion

In this paper, we introduce a modular and encoder-agnostic framework for unsupervised clustering of time-series data. By combining self-supervised pretraining with a cluster-friendly fine-tuning objective, our method enables effective latent space structuring without relying on labeled data. Experimental results on multiple real-world time-series datasets validate the effectiveness of our approach, showing clear improvements over clustering methods that use only pretrained embeddings without fine-tuning. While the proposed framework is designed to be compatible with a wide range of representation learning backbones, our current experiments focus exclusively on a TS2Vec-based encoder. As part of future work, we plan to extend our evaluation to include more diverse self-supervised paradigms such as representation learning based on large language models (LLMs), thereby further validating the generalizability and extensibility of our clustering approach.

Acknowledgments

This work was supported by the Instltute of Infomation & Communications Technology Planning & Evaluation(IITP) grant funded by the Korea govemment(MSIT)(RS-2024-00402782, Development of virtual commissioning technology that interconnects manufacturing data for global expansion)

References

[1] J. Dumler, S. Faatz, M. Friedrich, and F. Döpper, "Automatic time series segmentation and clustering for process monitoring in series production," *Procedia CIRP*, vol. 118, pp. 602–607, 2023.

[2] J. Wang, P. Liu, M. F. She, S. Nahavandi, and A. Kouzani, "Biomedical time series clustering based on non-negative sparse coding and probabilistic topic model," *Computer methods and programs in biomedicine*, vol. 111, no. 3, pp. 629–641, 2013.

[3] M. Ono, T. Katsuki, M. Makino, K. Haida, and A. Suzuki, "Interpretation method for continuous glucose monitoring with subsequence time-series clustering," in *Digital Personalized Health and Medicine*. IOS Press, 2020, pp. 277–281.

[4] Z. Yue, Y. Wang, J. Duan, T. Yang, C. Huang, Y. Tong, and B. Xu, "Ts2vec: Towards universal representation of time series," in *Proceedings of the AAAI conference on artificial intelligence*, vol. 36, no. 8, 2022, pp. 8980–8987.

[5] B. Yang, X. Fu, N. D. Sidiropoulos, and M. Hong, "Towards k-means-friendly spaces: Simultaneous deep learning and clustering," in *international conference on machine learning*. PMLR, 2017, pp. 3861–3870.

[6] Q. Ma, J. Zheng, S. Li, and G. W. Cottrell, "Learning representations for time series clustering," *Advances in neural information processing systems*, vol. 32, 2019.

[7] H. A. Dau, A. Bagnall, K. Kamgar, C.-C. M. Yeh, Y. Zhu, S. Gharghabi, C. A. Ratanamahatana, and E. Keogh, "The ucr time series archive," *IEEE/CAA Journal of Automatica Sinica*, vol. 6, no. 6, pp. 1293–1305, 2019.

[8] A. F. McDaid, D. Greene, and N. Hurley, "Normalized mutual information to evaluate overlapping community finding algorithms," *arXiv preprint arXiv:1110.2515*, 2011.

[9] W. M. Rand, "Objective criteria for the evaluation of clustering methods," *Journal of the American Statistical association*, vol. 66, no. 336, pp. 846–850, 1971.