

Learnable Matrix Profile for Motif Discovery on Multivariate Time Series

Mingkai Lin*, Yinke Wang*, Xiaobin Hong, Wenzhong Li†

State Key Laboratory for Novel Software Technology, Nanjing University, Nanjing, China
 {yinke_wang,xiaobinhong}@smail.nju.edu.cn, {mingkai,lwz}@nju.edu.cn

Abstract

Multivariate motif discovery aims to identify frequently occurring subsequences within multi-dimensional time series, which is a critical machine learning task with wide applications. However, previous motif discovery algorithms often miss complex multivariate motifs and struggle with high computational costs as data scale and dimensionality grow. We propose a novel Learnable MultivariAte matrix Profile method (L-MAP) that captures inter-dimensional dependencies for comprehensive analysis of multivariate time series. The time series is partitioned into subsequences using the Fourier transform in the frequency domain, with locality-sensitive hashing (LSH) assigning them to buckets based on distinct patterns. Each subsequence is modeled as a graph for multivariate fusion, where triplet learning is used to capture cross-dimensional relationships and form graph embeddings. Unlike prior methods relying on Euclidean distance modeling, our graph-based approach computes all-pairs similarity in a latent space, which constructs the multivariate matrix profile from distributions formed by embedding clusters. Extensive experiments on multivariate datasets from diverse domains demonstrate that L-MAP outperforms state-of-the-art methods in motif discovery, offering superior quality, diversity, and scalability efficiency.

Introduction

With the rapid development of modern information systems and sensors, vast amounts of data are now collected and stored as multivariate time series (MTS) (Hamilton 2020; Lim and Zohren 2021; Qiu et al. 2025). Motif discovery is a fundamental machine learning task that aims to identify frequently occurring patterns or subsequences in time series (Chiu, Keogh, and Lonardi 2003). These motifs often reflect the recurring behaviors or critical states of the system, while patterns deviating from them are recognized as discords or anomalies (Li et al. 2013; Torkamani and Lohweg 2017; Blázquez-García et al. 2021; Germain, Truong, and Oudre 2024). Over the past decades, motif discovery has proven useful in applications such as medical diagnosis (Liu et al. 2015; Kraljevska et al. 2025), financial market analy-

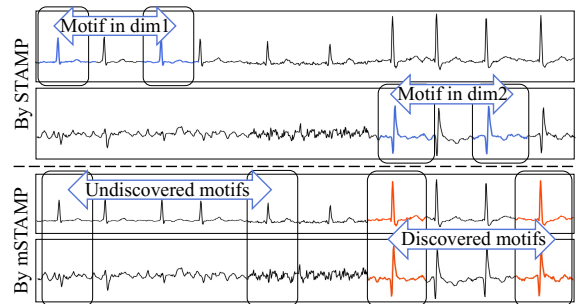


Figure 1: The case study in the AtrialFibrillation dataset

sis (Pradeepkumar and Ravi 2020; Zhang et al. 2025), and system maintenance (Stamatescu et al. 2022).

To efficiently extract motifs and discords from time series, the data structure of matrix profile (MP) was introduced to rapidly compute nearest neighbor distances between subsequences (Yeh et al. 2016). It provides crucial insights for analyzing time series with Low MP values indicating frequent patterns (motifs) and high values for anomalies (discords). Many MP-based algorithms have been proposed for diverse scenarios. For example, STAMP (Yeh et al. 2016) utilizes the Fourier transform to compute all-pairs similarity and generate profile values. MDMS (Zhu, Mueen, and Keogh 2019) supports motif discovery under missing values using lower bounds for distance computation. DAMP (Lu et al. 2022) supports fast and exact discord detection in streaming time series. In this way, MP has become a fundamental tool for more effective time series analysis, enabling fast and scalable computation to reveal key patterns in time series.

Despite these advances, most MP-based algorithms are designed for univariate data and do not generalize well to multivariate time series. A direct statistical aggregation of MP values across multivariate often obscures valuable dimensional information. Thus, methods such as mSTAMP (Yeh, Kavantzias, and Keogh 2017) have extended the vanilla STAMP method with a k -dimensional distance function to mine meaningful motifs through a multidimensional matrix profile. However, these extensions still face significant limitations in efficiently handling the complexity of multivariate temporal structures. To better illustrate the limitations intuitively, we conduct experiments on two representative MP-

*Equal contribution.

†The corresponding author is Wenzhong Li (lwz@nju.edu.cn).
 Copyright © 2026, Association for the Advancement of Artificial Intelligence (www.aaai.org). All rights reserved.

Length	5000	10000	20000	50000	100000
STAMP (vanilla MP)	95	269	834	6284	38421
mSTAMP	174	686	2658	17428	41783
Dimension	1	5	10	25	50
STAMP (vanilla MP)	270	1354	2656	6486	12932
mSTAMP	121	1663	3559	11360	28711

Table 1: Computation time of different algorithms with increasing length and dimension, recorded in seconds

based methods, STAMP and mSTAMP, as a case study.

Fig. 1 shows that applying STAMP independently to each dimension of the AtrialFibrillation dataset (Bagnall et al. 2018) leads to misaligned motif detections and in contrast, mSTAMP fails to detect motifs distorted by temporal shifts and amplitude variations, ignoring potentially essential patterns. Furthermore, as shown in Tab. 1, for the WADI (Ahmed, Palleti, and Mathur 2017) dataset, both STAMP and mSTAMP incur significantly higher computation times as the sequence length or dimensionality increases, rendering them inefficient for real-time multivariate analysis on high-dimensional data streams. Therefore, there are three significant challenges remaining:

- (1) **Misalignment of identified motifs:** applying MP-based methods to individual dimensions of a time series could lead to discrepancies among the recovered patterns.
- (2) **Overlooking important patterns:** MP-based methods may omit potentially significant motifs due to temporal shifts and amplitude variations on multiple dimensions.
- (3) **Inefficiency with multivariate time series:** existing MP-based algorithms face significant efficiency challenges, especially with increasing data volume and dimensionality.

To address the challenges, we propose a novel Learnable Multivariate Ate Matrix Profile method, termed **L-MAP**, designed for efficient and scalable motif discovery in multivariate time series, with three modules as shown in Fig. 2.

In the first *pattern extraction* module, the input time series is segmented into subsequences, and their frequency-domain representations are obtained via the Fourier transform. We apply locality-sensitive hashing (LSH) to group similar patterns into buckets, enabling efficient organization of subsequences. In the second *multivariate fusion* module, each multivariate subsequence is modeled as a graph, where nodes represent individual dimensions and edges reflect their interdependencies. Using LSH buckets, we construct triplets consisting of anchor, positive, and negative samples. A trainable graph encoder is then applied to learn latent embeddings via triplet loss, capturing both intra- and inter-dimensional structures. In the final *profile construction* module, rather than relying on Euclidean distance, we measure all-pairs similarity in the embedding space using graph-based representations. A learnable multivariate matrix profile is constructed by estimating distributional similarities between embedding clusters, enabling robust motif and discord discovery under complex, high-dimensional patterns.

Our main contributions can be summarized as follows:

- We propose L-MAP, a novel learnable multivariate matrix profile framework that models inter-dimensional re-

lationships in latent space, overcoming limitations of traditional MP methods.

- We design a Fourier-based LSH scheme for efficient and linear-time pattern extraction, effectively grouping subsequences with similar frequency characteristics.
- We introduce a graph-based modeling approach using triplet learning to derive expressive representations of multivariate subsequences, enabling precise similarity measurement beyond raw distance metrics.
- We conduct experiments on multivariate datasets from diverse domains, demonstrating that L-MAP outperforms state-of-the-art methods in motif discovery, offering superior quality, diversity, and scalability efficiency.

Related Work

Most matrix profile (MP) based algorithms address problems related to univariate time series data, identifying motifs or anomalies through the generated profile values. The classical matrix profile algorithm STAMP (Yeh et al. 2016) evaluated the profile values using the MASS algorithm (Zhong and Mueen 2024), which utilized the Fast Fourier Transform to calculate each dot product between two subsequences. In order to improve the efficiency of MP generation, STOMP (Zhu et al. 2016) was then proposed, which combined MP generation with GPU calculation that significantly enhanced the scalability of motif discovery. To further improve processing speed, (Zhu et al. 2018) introduced SCRIMP++, an $O(n^2)$ time algorithm that enabled real-time, interactive motif discovery. To address missing data, the MDMS (Zhu, Mueen, and Keogh 2019) algorithm utilized lower bounding functions for the Euclidean distance in matrix profile computation, enabling motif discovery with missing values.

To extend the MP generation algorithm to handle stream data, a learning-based method LAMP (Zimmerman et al. 2019) was proposed to predict the univariate matrix profile. This method trained the model offline and generates MP values for incoming sequences. Due to the poor scalability of many anomaly detection algorithms for large-scale datasets, DAMP (Lu et al. 2022) quickly generated matrix profile values, thereby extracting discords from massive time series data. The work (Imamura and Nakamura 2024) proposed Spikelets and introduced the first lower bound for DTW in the Spikelet space, which efficiently identified DTW motif pairs of different lengths. For multivariate time series analysis, (Yeh, Kavantzias, and Keogh 2017) proposed mSTAMP, which mined meaningful motifs through a proposed multidimensional matrix profile. LAMA (Schäfer and Leser 2025) is the latest method for efficient multidimensional leitmotif discovery. For the task of multivariate time series anomaly detection, (Tafazoli and Keogh 2023) proposed the TSADIS algorithm, which extended the univariate matrix profile to multi-dimensional forms for quickly finding the best K of N anomaly subsets in the given MTS. Contrasting with previous algorithms, our work introduces a novel learnable multivariate matrix profile approach termed L-MAP, which is capable of capturing interdependencies across multiple dimensions and measuring all-pairs similarity within a latent space with scalability to large data volume and dimensions.

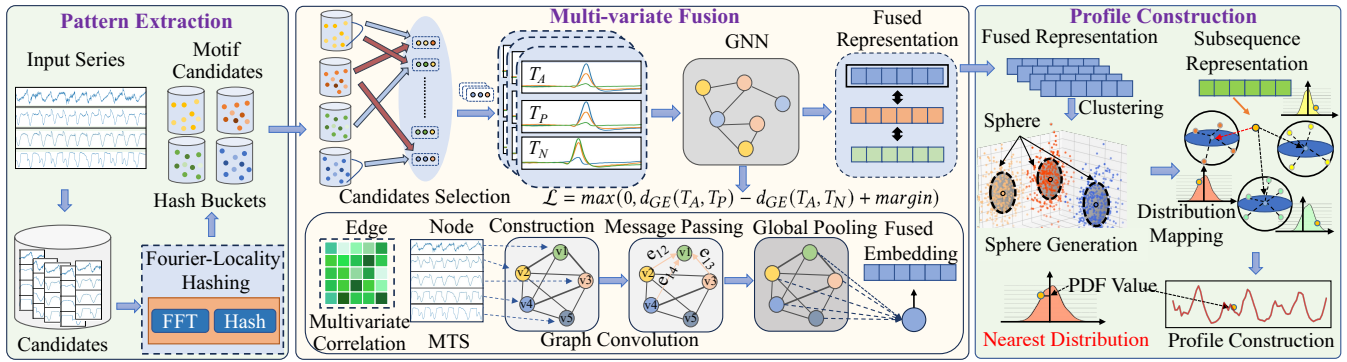


Figure 2: The overall framework of L-MAP.

Methodology

Notations and Definitions

Formally, a time series T is a sequence of data points arranged in chronological order, which is denoted as $\mathbf{x} = \{\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_N\}$, where $\mathbf{x}_t \in R^D (1 \leq t \leq N)$ is a D -dimensional vector. Based on the value of D , we can divide time series into univariate time series and multivariate time series, where $D = 1$ is univariate and $D > 1$ means multivariate time series (MTS). We then introduce the following notations and definitions for the problem formulation.

Definition 1 (Subsequence $T_{a,m}$) Given a time series T , a subsequence $T_{a,m}$ is a segment taken from original T , denoted as $T_{a,m} = \{\mathbf{x}_a, \mathbf{x}_{a+1}, \dots, \mathbf{x}_{a+m-1}\}, 1 \leq a \leq a+m-1 \leq N$, where a represents the start position and m denotes the length of this segment. Note that for multivariate time series, the subsequence is also a multivariate subsequence.

Definition 2 (Distance d) The distance measure d assesses the difference between two given subsequences $T_{p,m}$ and $T_{q,m}$ with length m , which can be calculated by their L_2 norm:

$$d(T_{p,m}, T_{q,m}) = \sqrt{\sum_{i=0}^{m-1} (\mathbf{x}_{p+i} - \mathbf{x}_{q+i})^2}. \quad (1)$$

Definition 3 (Nearest Neighbor nn) Given a subsequence set $\mathbf{T}_m = \{T_{p,m}, T_{q,m}, \dots\}$ composed of subsequences of length m , the nearest neighbor of $T_{i,m} \in \mathbf{T}_m$ is the subsequence having the smallest distance to it, denoted by

$$nn(T_{i,m} : \mathbf{T}_m) = \arg \min_{T_{j,m} \in \mathbf{T}_m, j \neq i} d(T_{i,m}, T_{j,m}). \quad (2)$$

Definition 4 (Nearest Neighbor Distance d_n) Given a subsequence set $\mathbf{T}_m = \{T_{p,m}, T_{q,m}, \dots\}$ composed of subsequence samples of length m , the nearest neighbor distance of $T_{i,m} \in \mathbf{T}_m$ can be calculated by

$$d_n(T_{i,m} : \mathbf{T}_m) = \min_{T_{j,m} \in \mathbf{T}_m, j \neq i} d(T_{i,m}, T_{j,m}). \quad (3)$$

In conventional matrix profile algorithms, the L_2 norm or DTW (Alaee, Kamgar, and Keogh 2020) is used to measure subsequence distance (Yeh et al. 2016), and the nearest is used to form the matrix profile value as follows.

Definition 5 (Matrix Profile MP) Given a time series T and subsequence length m , a matrix profile $MP(T, m)$ is a vector of distance to annotate T , which is formulated as:

$$MP(T, m) = (mp_1, mp_2, \dots, mp_{|T|-m+1}), \quad (4)$$

$$mp_i = d_n(T_{i,m} : \mathbf{T}_m), \quad \mathbf{T}_m = \{T_{1,m}, T_{2,m}, \dots, T_{|T|-m+1}\},$$

where \mathbf{T}_m denotes the subsequence set generated by time series T with stride length 1 and $|T|$ denotes the total length of T . Each element mp_i represents the nearest neighbor distance of the specific subsequence in the overall time series.

Definition 6 (Profile Index PI) Given a time series T and subsequence length m , a profile index $PI(T, m)$ is:

$$PI(T, m) = (pi_1, pi_2, \dots, pi_{|T|-m+1}), \quad pi_i = nn(T_{i,m} : \mathbf{T}_m), \quad \mathbf{T}_m = \{T_{1,m}, T_{2,m}, \dots, T_{|T|-m+1}\}, \quad (5)$$

where each element pi_i in the profile index denotes the nearest neighbor of the specific subsequence at each timestamp.

The matrix profile enables motif discovery by identifying the smallest values and retrieving the corresponding subsequences via the profile index. Conversely, the largest values reveal discords, highlighting anomalous patterns. These motifs and discords help underpin key tasks. Different from traditional MP computation, our method introduces a learnable framework, L-MAP, that evaluates relationships between multivariate subsequences in both the frequency domain and latent space. As illustrated in Fig. 2, L-MAP follows a three-stage pipeline: *pattern extraction*, *multivariate fusion*, and *profile construction*. First, the input time series is segmented into subsequences and transformed via the Fourier transform. Locality-sensitive hashing (LSH) then groups subsequences into buckets based on frequency-domain patterns. In the fusion stage, each subsequence is modeled as a graph, and triplet learning captures inter-dimensional relationships through graph embeddings. Finally, in the profile construction stage, similarity is computed in the latent space by using distances between graph embedding clusters, which forms the multivariate matrix profile for downstream analysis. Further details are provided in the following three sections.

Pattern Extraction

Firstly, we utilize a sliding window to partition the original time series into a set of subsequent samples.

Definition 7 (Time series partition) Given a time series $T \in R^{N \times D}$, a sliding window length w and a sliding stride s , the time series partition operation divides the time series into a set of overlapping subsequences denoted as $S = \{S_1, S_2, \dots, S_{\lfloor \frac{N-w}{s} \rfloor + 1}\}$, where $S_i = T_{i,w} \in R^{w \times D}$ is a subsequence as in Definition 1.

After applying time series partition, we perform Fourier transform on the decomposed multivariate subsequences and extract the amplitude as internal information from the frequency domain. The transforms are formulated as:

$$\begin{aligned} Z_{i,k} &= \text{Amp}(FFT(S_{i,k})), \\ Z_i &= \text{concat}(Z_{i,1}, Z_{i,2}, \dots, Z_{i,D}), \quad \mathcal{Z} = \{Z_1, Z_2, \dots, Z_{|S|}\}, \end{aligned} \quad (6)$$

where $S_{i,k}$ denotes the univariate series in the k -th channel of the subsequence S_i ; $FFT(\cdot)$ represents the Fourier transform, and $\text{Amp}(\cdot)$ denotes the amplitude calculation of the signals in the frequency domain.

The generated components of all dimensions are then concatenated into a vector Z_i , serving as a new representation for the original sequence S_i which captures the inner information of the original subsequence. Focusing on the frequency patterns can eliminate misalignment across different dimensions while preserving the essential features of subsequences. Once we have obtained the frequency representations of these subsequences, the distance between each pair of representations is employed as a metric for allocation and filtering through the locality-sensitive hashing.

Locality-sensitive hashing (LSH) (Gionis et al. 1999) is a hashing-based approach that can map different subsequences into the corresponding buckets with linear complexity. The definition of the proposed LSH family \mathcal{F} is denoted as below. Let (M, d) be a metric space and let the threshold $r > 0$ and an approximation factor $c > 1$. For any frequency representation $Z_i, Z_j \in A$ and a hash function h chosen uniformly at random from \mathcal{F} , such a family of hash functions is called $(r, cr, p1, p2)$ -sensitive if it satisfies the conditions:

$$\begin{aligned} \text{if } d(Z_i, Z_j) < r, \Pr_h[h(Z_i) = h(Z_j)] &\geq p_1, \\ \text{if } d(Z_i, Z_j) \geq cr, \Pr_h[h(Z_i) = h(Z_j)] &\leq p_2, \quad \text{s.t. } p_1 > p_2 \end{aligned} \quad (7)$$

where $h(\cdot)$ represents the hash function that maps each representation into a bucket, and \Pr_h represents the probability function, i.e., the probability that Z_i and Z_j are mapped to the same bucket. For distance measuring, we employ the L_2 norm between two representations. With LSH, time series with similar representations are mapped into the same bucket with high probability, whereas subsequences falling into different buckets indicate distinct patterns.

Multivariate Fusion

By examining the distances between samples selected by the buckets, we can develop a model that captures the interrelationships among multivariates. Specifically, we propose a graph learning approach for multivariate fusion and measure pair-wise distances in the graph embedding space. We formulate the multivariate subsequences as a graph as follows.

Definition 8 (Time Series Graph G) Given a subsequence $T_{i,m} \in R^{m \times D}$, where D denotes the dimensionality of the

original MTS, the time series graph $G_{T_{i,m}}$ of $T_{i,m}$ can be denoted as,

$$G_{T_{i,m}} = (X_{i,m}, A), \quad X_{i,m} = (T_{i,m}^1, T_{i,m}^2, \dots, T_{i,m}^D), \quad (8)$$

where each univariate subsequence $T_{i,m}^k \in R^m$ in $T_{i,m}$ is modeled as a node in the graph with feature $X_{i,m} \in R^{D \times m}$; and $A \in R^{D \times D}$ is an adjacency matrix representing the relationships between dimensions in the MTS, i.e., a fully connected graph with all elements in the matrix equal to 1.

Definition 9 (Graph Embedding Distance d_{GE}) Given two subsequences $T_{p,m}$ and $T_{q,m}$ of length m , the graph embedding distance between them can be denoted as

$$d_{GE}(T_{p,m}, T_{q,m}) = \|E_G(G_{T_{p,m}}) - E_G(G_{T_{q,m}})\|_2, \quad (9)$$

where E_G denotes the graph encoding function, which transforms a given graph into a latent representation.

In this work, we use a graph neural network (GNN) (Zhou et al. 2020; Lin et al. 2022, 2025) as the graph encoder. In this way, we can effectively model temporal relationships across different dimensions, allowing for a more comprehensive calculation of interactions among multivariates. We utilize triplet learning (Jaiswal et al. 2020) to adequately learn the interior information of the given subsequences. For each triplet, we first select an arbitrary subsequence as an *anchor*. Then, we consider the subsequences with the closest frequency domain distances within the same bucket as *positive sample*, and randomly choose a subsequence from different buckets as the *negative sample*, thus forming a triplet formulated as:

$$\mathcal{T} = (T_A, T_P, T_N), \quad (10)$$

where T_A , T_P , and T_N respectively denote the anchor, positive, and negative subsequence through selection.

In the graph convolutional network, we first utilize a learnable linear layer to project all the node features in the time series graph of $T_{i,m}$ into the same dimensional space:

$$H_i^0 = \text{Proj}_{\text{linear}}(G_{T_{i,m}}), \quad H_i^0 = (h_1^0, h_2^0, \dots, h_D^0), \quad (11)$$

where the number of elements in H_i^0 equals the dimensionality D of the original time series. This hidden representation of nodes serves as the initial state of the time series graph, and we utilize a fully connected graph to establish connections between nodes in the graph, denoted as an adjacency matrix $A \in R^{D \times D}$. The propagation function of graph learning can be denoted as:

$$H^{l+1} = f(H^l, A), \quad f(H^l, A) = \sigma(\hat{D}^{-\frac{1}{2}} \hat{A} \hat{D}^{-\frac{1}{2}} H^l W^l), \quad (12)$$

where H^l represents the l^{th} hidden state of the graph, \hat{D} is the degree matrix of \hat{A} , and $\hat{A} = A + I$ where I represents an identity matrix $\in R^{D \times D}$. After multiple layers of graph propagation, the model can thoroughly learn the mutual relationships between multiple variables, and the hidden states of each node can encapsulate a wealth of information. Finally, we utilize mean graph pooling to fuse the learned representations of all nodes into an integrated vector, serving as the graph representation of the original input subsequence,

$$E_G(G_{T_{i,m}}) = \frac{1}{D} \sum_{i=1}^D h_i^L, \quad (13)$$

where h_i^L represents the hidden state of node i in the graph hidden state H_i^L after L layers of propagation.

Based on the above settings, the triplet loss function of the graph model is given as follows,

$$\mathcal{L} = \max(0, d_{GE}(T_A, T_P) - d_{GE}(T_A, T_N) + \text{margin}), \quad (14)$$

where T_A , T_P , and T_N respectively represent the anchor subsequence, positive sample, and negative sample selected via the locality-sensitive hashing buckets; d_{GE} represents the graph embedding distance introduced in Definition 9, where we utilize a GNN to encode the given subsequences.

Profile Construction

After obtaining the representations of the subsequences, we can utilize the representation of each subsequence to construct the matrix profile values. Firstly, given the representations fused by the graph model, we apply clustering methods to group all the embeddings into several clusters. For each cluster center, we define a hypersphere with a specific radius to construct a high-dimensional spherical space. Each point inside the sphere generates a smooth distribution based on a parameter-free probability density function. This allows us to create distinct sphere distributions for different clusters, representing various types of motifs. For each representation generated by a subsequence, we calculate its probability density within each sphere distribution. The distribution with the highest probability density is selected as the nearest distribution for the subsequence, and the probability density value within this distribution is then used to generate the matrix profile value.

Firstly, given a set E that contains the representations fused by the graph model, we apply a clustering method to generate several cluster centers in the high-dimensional embedding space. For each cluster center, we construct a high-dimensional sphere $Q(c, r)$ to generate a distribution for further inference.

Definition 10 (High-dimensional sphere) Given the cluster center c and a sphere radius r , the high-dimensional sphere $Q(c, r)$ can be denoted as:

$$Q(c, r) = \{y \mid \|y - c\|_2 \leq r\}, \quad (15)$$

which means that a representation with embedding distances less than the sphere radius r from the center c is considered as an instance of the current sphere distribution.

We adopt the kernel density estimation to fit the spherical distributions. For each embedding, we calculate the density estimation value for each sphere distribution, selecting the one with the maximum density as its nearest distribution and generating the profile mapping distance.

Definition 11 (Nearest Distribution) Given an embedding e_i and the set of sphere distributions $\mathcal{D} = \{\mathcal{D}_1, \mathcal{D}_2, \dots, \mathcal{D}_P\}$, the nearest distribution nd can be denoted as:

$$nd(e_i : \mathcal{D}) = \arg \max_{\mathcal{D}_i \in \mathcal{D}} \left(\frac{1}{|\mathcal{D}_i|b} \sum_{c \in \mathcal{D}_i} K\left(\frac{\|e_i - c\|_2}{b}\right) \right), \quad (16)$$

where P represents the number of distributions; c represents each component in a specific distribution \mathcal{D}_i ; $K(\cdot)$ represents the selected kernel function and b denotes the bandwidth in the kernel function; $|\mathcal{D}_i|$ represents the number of elements in distribution \mathcal{D}_i .

Definition 12 (Profile Mapping Distance) Given an embedding e_i and sphere distribution set $\mathcal{D} = \{\mathcal{D}_1, \mathcal{D}_2, \dots, \mathcal{D}_P\}$, the profile mapping distance d_{pm} can be calculated as:

$$d_{pm}(e_i : \mathcal{D}) = \min_{\mathcal{D}_i \in \mathcal{D}} \left(1 - \frac{1}{|\mathcal{D}_i|b} \sum_{c \in \mathcal{D}_i} K\left(\frac{\|e_i - c\|_2}{b}\right) \right), \quad (17)$$

where the symbols in the formula have the same meanings as defined in Definition 11.

With the above information, we formulate the learnable matrix profile L-MAP and L-MAP index as follows.

Definition 13 (Learnable Matrix Profile L-MAP)

Given a time series T , subsequence length m , the graph encoding function E_G and the sphere distributions $\mathcal{D} = \{\mathcal{D}_1, \mathcal{D}_2, \dots, \mathcal{D}_P\}$, the learnable matrix profile L-MAP(T, m) can be formulated as:

$$L\text{-MAP}(T, m) = (\text{map}_1, \text{map}_2, \dots, \text{map}_{|T|-m+1}), \quad (18)$$

$$\text{map}_i = d_{pm}(e_i : \mathcal{D}), \quad e_i = E_G(G_{T_{i,m}}),$$

where $G_{T_{i,m}}$ denotes the time series graph transformed by subsequence $T_{i,m}$, which is defined in Definition 8.

Definition 14 (L-MAP Index) Given a time series T , subsequence length m , the graph encoding function E_G and the sphere distributions $\mathcal{D} = \{\mathcal{D}_1, \mathcal{D}_2, \dots, \mathcal{D}_P\}$, the L-MAP Index MPI can be formulated as

$$MPI(T, m) = (\text{mpi}_1, \text{mpi}_2, \dots, \text{mpi}_{|T|-m+1}), \quad (19)$$

$$\text{mpi}_i = nd(e_i : \mathcal{D}), \quad e_i = E_G(G_{T_{i,m}}),$$

where $G_{T_{i,m}}$ denotes the time series graph transformed by subsequence $T_{i,m}$, which is defined in Definition 8.

A lower value in the matrix profile suggests multiple subsequences in the entire time series that closely resemble its pattern, and it can be considered a motif. A higher value indicates that for the entire segment of the time series, it does not fit well into any distribution, and can be viewed as discords. Finally, the profile values are arranged based on the timestamp of each embedding, forming the L-MAP and its index annotated to the original time series.

Experiments

Experimental Settings

We evaluate the effectiveness of our approach via 15 MTS classification datasets from the UEA MTSC archive (Baganall et al. 2018), a large multivariate dataset covering various domains. The information on the utilized classification datasets is shown in Tab. 2. We compare six baselines: (1) STOMP (Zhu et al. 2016): a method leveraging GPU acceleration for matrix profile computation. (2) Multi-MP: an extension of STOMP that computes matrix profiles independently per dimension and aggregates them column-wise. (3) SCRIMP++ (Zhu et al. 2018): an $O(n^2)$ algorithm enabling interactive motif discovery. (4) DAMP (Lu et al. 2022): a scalable matrix profile-based method for discord discovery, compared in terms of time consumption. (5) mSTAMP (Yeh, Kavantzias, and Keogh 2017): a multivariate matrix profile approach for discovering cross-dimensional motifs. (6)

Datasets	Information				Classification Results					
	Type	Dim.	Len.	Cl.	STOMP	SCRIMP++	LAMA	Multi-MP	mSTAMP	L-MAP
ArticulatoryWordRecognition	MOTION	9	144	25	0.970	0.973	0.967	0.983	0.980	0.990
AtrialFibrillation	ECG	2	640	3	0.267	0.333	0.267	0.400	0.533	0.533
BasicMotions	HAR	6	100	4	1.000	1.000	1.000	1.000	1.000	1.000
CharacterTrajectories	MOTION	3	116	20	0.978	0.975	0.975	0.969	0.978	0.989
Cricket	HAR	6	1197	12	0.986	0.972	0.917	0.972	0.958	0.986
Epilepsy	HAR	3	206	4	0.942	0.978	0.884	0.862	0.906	0.986
ERing	HAR	4	65	6	0.870	0.915	0.707	0.848	0.804	0.948
FingerMovements	EEG	28	50	2	0.570	0.630	0.570	0.590	0.520	0.640
Handwriting	HAR	3	152	26	0.373	0.378	0.167	0.284	0.313	0.452
Heartbeat	AUDIO	61	405	2	0.761	0.766	0.722	0.746	0.751	0.766
LSST	OTHER	6	36	14	0.581	0.584	0.481	0.536	0.534	0.586
NATOPS	HAR	24	51	6	0.917	0.922	0.678	0.767	0.806	0.905
RacketSports	HAR	6	30	4	0.783	0.803	0.651	0.671	0.658	0.816
SelfRegulationSCP2	EEG	7	1152	2	0.589	0.578	0.500	0.567	0.561	0.572
StandWalkJump	ECG	4	2500	3	0.333	0.400	0.533	0.333	0.333	0.533

Table 2: Classification datasets information and the motif quality comparison

LAMA (Schäfer and Leser 2025): the method for efficient multidimensional leitmotif discovery, configured to search for motifs across the entire dimensional space. All results are averaged over five runs. Experiments are implemented in PyTorch (Python 3.6.8) and conducted on a machine with an Intel Xeon E5-2620 v2 CPU (2.10GHz), 64GB RAM, a GeForce RTX 2070 (8GB), and 64-bit CentOS Linux 7.2.

Numerical Results

We present experiments to evaluate the effectiveness and efficiency of the L-MAP framework from three aspects: *Motif Quality*, *Motif Diversity* and *Scalability Efficiency*.

[Motif Quality] We evaluate the motif quality via MTS classification accuracy according to (Lines et al. 2012). Effective multivariate motifs are characterized by their frequency and ability to capture meaningful, recurring patterns across multiple dimensions. Unlike shapelets (Le et al. 2024), which are explicitly selected for class discrimination, motifs contribute to classification by providing their structural and contextual information inherent in the data. Thus, high accuracy reflects the informativeness, representativeness, and cross-dimensional coherence of the discovered motifs. We show the results in Tab. 2. It’s obvious that L-MAP outperforms other methods in time series classification accuracy on the vast majority of datasets. Although our method is not specifically designed for classification, L-MAP still achieves better classification performance than the baselines. Since STOMP and SCRIMP++ rely solely on motifs in a single dimension, their performance is relatively poor on most datasets. While Multi-MP, mSTAMP and LAMA account for multivariate computations, they overlook the inter-variable relationships within time series, leading to generally unsatisfactory classification results. The L-MAP model can effectively learn the relationships between dimensions in features, thus identifying motifs with better quality.

[Motif Diversity] We visualize the discovered motifs on three datasets in Fig. 3 to compare L-MAP with mSTAMP and LAMA, two effective motif discovery methods for MTS. In each subfigure, the upper shows the input time series color-coded by L-MAP motif clusters, followed by the matrix profile values and indices via L-MAP and mSTAMP; the lower marks the motif positions identified by LAMA.

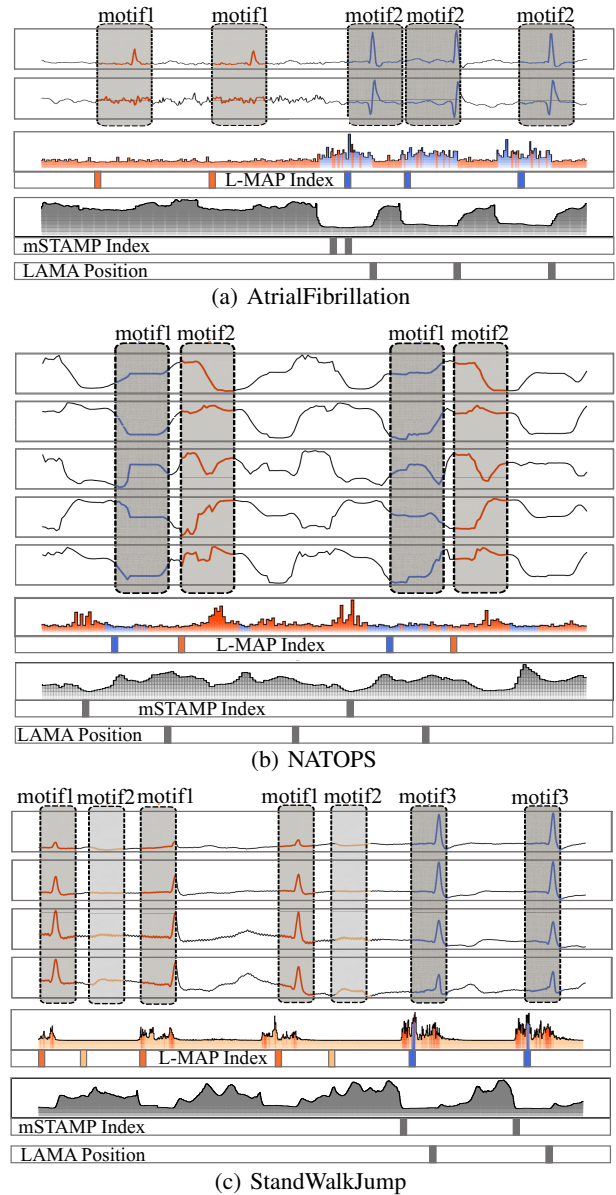


Figure 3: Visualization of the discovered motifs

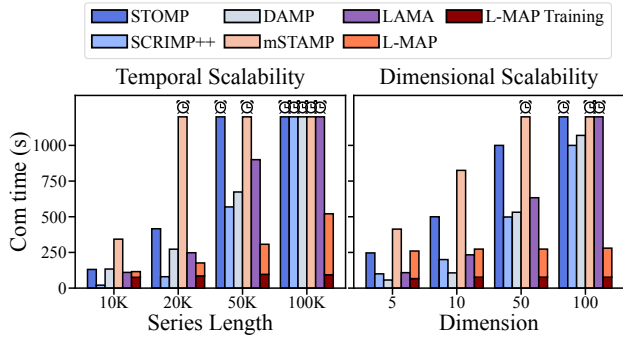


Figure 4: Computation time of different algorithms.

AtrialFibrillation is a two-dimensional ECG dataset. L-MAP captures two distinct motif types (orange and blue), including both low-amplitude patterns on the left and larger motifs on the right. In contrast, mSTAMP and LAMA only detect the prominent high-amplitude motifs, missing the subtle ones. NATOPS is a 24-dimensional dataset from hand and arm sensors; five dimensions are shown for clarity. Despite temporal misalignment in dimensions 3 and 4, L-MAP identifies two motif types from different clusters. mSTAMP and LAMA, however, fail to capture the misaligned (orange) motif. StandWalkJump includes ECG recordings during physical activity. In the standing segment, L-MAP discovers three motif types: low-amplitude (orange), high-amplitude multi-dimensional (blue), and stable across dimensions (yellow). mSTAMP and LAMA only detect a subset, showing limited diversity. These results demonstrate L-MAP’s effectiveness in handling high-dimensional data, modeling cross-dimensional relationships, and discovering diverse motifs even under variations in scale and alignment.

[Scalability Efficiency] We calculated the time cost of different approaches by selecting subsets of various lengths and dimensions in the WADI dataset (Ahmed, Palleti, and Mathur 2017) for validation. The results are depicted in Fig. 4, where the clock symbol on the bar chart indicates a timeout. According to the figure, the time cost of calculating profile values for different algorithms increases with the length and dimension of the time series. Despite the acceleration from GPU training, STOMP and mSTAMP still incur excessive time costs for longer input sequences. SCRIMP++ maintains high efficiency with $O(n^2)$ complexity when the data size is not large, but it incurs timeouts when dealing with larger datasets. DAMP and LAMA perform well for small dimensions, but their cost increases rapidly with larger lengths and dimensions. L-MAP exhibits scalability in handling extremely long MTS data and achieves a significant reduction in computational time compared to the baselines.

Ablation and Analysis

[Ablation Study] In this section, we conduct comparison experiments on classification datasets to verify the effectiveness of fast Fourier transform (FFT) and graph convolution (GC). We follow the same procedure as in previous classification tasks, using motifs identified by L-MAP to perform

Metric	Accuracy		
	w/o FFT	w/o GC	L-MAP
ERing	0.922	0.944	0.948
Epilepsy	0.956	0.949	0.986
AtrialFibrillation	0.467	0.467	0.533
FingerMoveMents	0.570	0.590	0.640
Handwriting	0.434	0.427	0.452
Heartbeat	0.751	0.761	0.766

Table 3: Ablation study results.

classification on different datasets. As illustrated in Tab. 3, either removing the Fourier Transform or the graph convolution would lead to a reduction in the quality of the discovered motifs, which in turn results in decreased classification performance. The results confirm the positive impact of these two modules in L-MAP for multivariate motif discovery.

[Hyperparameter Analysis] In Fig. 5, we evaluate the impact of different parameters in L-MAP. We conduct experiments on three classification datasets, modifying the motif length m , hash size h , and sphere radius r to examine the influence of these hyperparameters. The values for m_1, m_2, m_3, m_4, m_5 are set to 20, 30, 40, 50, 60 for ERing and CharacterTrajectories, and 25, 50, 100, 150, 200 for Epilepsy, respectively. As shown in the figures, the performance improves with the increase in motif length. However, when the motif length exceeds a certain threshold, the performance starts to decline. Besides, the performance of L-MAP remains relatively stable across datasets as the hash size varies. Lastly, the performance of L-MAP is generally stable with changes in the sphere radius, but the performance slightly decreases when the radius exceeds 0.7.

Conclusion

Previous motif discovery methods often miss multivariate patterns and suffer from high computational costs as the data scale grows. We present L-MAP, a novel learnable multivariate matrix profile approach for efficient time series analysis. L-MAP uses Fourier-based LSH to group subsequences, models cross-dimensional dependencies via graph embeddings, and computes similarity in the latent space to construct the matrix profile. Extensive experiments demonstrate its effectiveness in discovering multivariate motifs and achieving strong performance across various datasets.

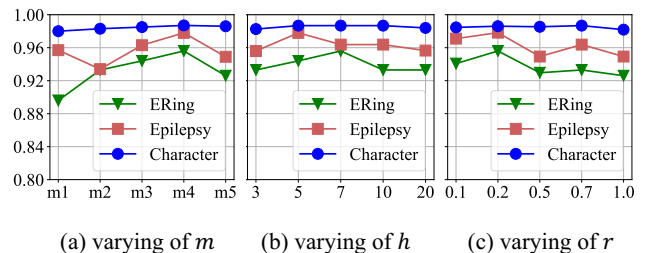


Figure 5: Hyperparameter analysis on classification datasets.

Acknowledgments

This work was supported by Basic Research Program of Jiangsu (Grant No. BK20251198), the Natural Science Foundation of Jiangsu Province (Grant No. BK20222003), the National Natural Science Foundation of China (Grant Nos. 62502201, 62572236), Postdoctoral Fellowship Program of CPSF (Grant No. GZC20251067), Jiangsu Funding Program for Excellent Postdoctoral Talent, the Collaborative Innovation Center of Novel Software Technology and Industrialization, and the Sino-German Institutes of Social Computing. We thank Professor Eamonn Keogh and his team for their pioneering work on the Matrix Profile and insightful perspectives on time series analysis.

References

- Ahmed, C. M.; Palleti, V. R.; and Mathur, A. P. 2017. WADI: a water distribution testbed for research in the design of secure cyber physical systems. In *Proceedings of the 3rd international workshop on cyber-physical systems for smart water networks*, 25–28.
- Alaee, S.; Kamgar, K.; and Keogh, E. 2020. Matrix profile XXII: exact discovery of time series motifs under DTW. In *2020 IEEE International Conference on Data Mining (ICDM 2020)*, 900–905. IEEE.
- Bagnall, A.; Dau, H. A.; Lines, J.; Flynn, M.; Large, J.; Bostrom, A.; Southam, P.; and Keogh, E. 2018. The UEA multivariate time series classification archive, 2018. *arXiv preprint arXiv:1811.00075*.
- Blázquez-García, A.; Conde, A.; Mori, U.; and Lozano, J. A. 2021. A review on outlier/anomaly detection in time series data. *ACM Computing Surveys (CSUR)*, 54(3): 1–33.
- Chiu, B.; Keogh, E.; and Lonardi, S. 2003. Probabilistic discovery of time series motifs. In *Proceedings of the ninth ACM SIGKDD international conference on Knowledge discovery and data mining (KDD 2003)*, 493–498.
- Germain, T.; Truong, C.; and Oudre, L. 2024. Persistence-based motif discovery in time series. *IEEE Transactions on Knowledge and Data Engineering (TKDE)*, 36(11): 6814–6827.
- Gionis, A.; Indyk, P.; Motwani, R.; et al. 1999. Similarity search in high dimensions via hashing. In *Proceedings of the 25th International Conference on Very Large Data Bases (VLDB 1999)*, volume 99, 518–529.
- Hamilton, J. D. 2020. *Time series analysis*. Princeton university press.
- Imamura, M.; and Nakamura, T. 2024. Efficient Discovery of Time Series Motifs under both Length Differences and Warping. In *Proceedings of the 30th ACM SIGKDD Conference on Knowledge Discovery and Data Mining (KDD 2024)*, 1188–1198.
- Jaiswal, A.; Babu, A. R.; Zadeh, M. Z.; Banerjee, D.; and Makedon, F. 2020. A survey on contrastive self-supervised learning. *Technologies*, 9(1): 2.
- Kraljevská, M.; Hlaváčková-Schindler, K.; Mikloutz, L.; and Plant, C. 2025. Motif Discovery Framework for Psychiatric EEG Data Classification. *arXiv preprint arXiv:2501.04441*.
- Le, X.-M.; Luo, L.; Aickelin, U.; and Tran, M.-T. 2024. Shapeformer: Shapelet transformer for multivariate time series classification. In *Proceedings of the 30th ACM SIGKDD Conference on Knowledge Discovery and Data Mining (KDD 2024)*, 1484–1494.
- Li, G.; Bräysy, O.; Jiang, L.; Wu, Z.; and Wang, Y. 2013. Finding time series discord based on bit representation clustering. *Knowledge-Based Systems*, 54: 243–254.
- Lim, B.; and Zohren, S. 2021. Time-series forecasting with deep learning: a survey. *Philosophical Transactions of the Royal Society A*, 379(2194): 20200209.
- Lin, M.; Hong, X.; Li, W.; and Lu, S. 2025. Unified Graph Neural Networks Pre-training for Multi-domain Graphs. In *Proceedings of the AAAI Conference on Artificial Intelligence (AAAI 2025)*, volume 39, 12165–12173.
- Lin, M.; Li, W.; Li, D.; Chen, Y.; and Lu, S. 2022. Resource-efficient training for large graph convolutional networks with label-centric cumulative sampling. In *Proceedings of the ACM Web Conference (WWW 2022)*, 1170–1180.
- Lines, J.; Davis, L. M.; Hills, J.; and Bagnall, A. 2012. A shapelet transform for time series classification. In *Proceedings of the 18th ACM SIGKDD international conference on Knowledge discovery and data mining (KDD 2012)*, 289–297.
- Liu, B.; Li, J.; Chen, C.; Tan, W.; Chen, Q.; and Zhou, M. 2015. Efficient motif discovery for large-scale time series in healthcare. *IEEE Transactions on Industrial Informatics*, 11(3): 583–590.
- Lu, Y.; Wu, R.; Mueen, A.; Zuluaga, M. A.; and Keogh, E. 2022. Matrix profile XXIV: scaling time series anomaly detection to trillions of datapoints and ultra-fast arriving data streams. In *Proceedings of the 28th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD 2022)*, 1173–1182.
- Pradeepkumar, D.; and Ravi, V. 2020. Financial time series prediction: an approach using motif information and neural networks. *International Journal of Data Science*, 5(1): 79–109.
- Qiu, X.; Wu, X.; Lin, Y.; Guo, C.; Hu, J.; and Yang, B. 2025. Duet: Dual clustering enhanced multivariate time series forecasting. In *Proceedings of the 31st ACM SIGKDD Conference on Knowledge Discovery and Data Mining (KDD 2025)*, 1185–1196.
- Schäfer, P.; and Leser, U. 2025. Discovering Leitmotifs in Multidimensional Time Series. *Proceedings of the VLDB Endowment*, 18(2): 377–389.
- Stamatescu, G.; Plamanescu, R.; Dumitrescu, A.-M.; Ciomei, I.; and Albu, M. 2022. Multiscale data analytics for residential active power measurements through time series data mining. In *2022 IEEE 7th International Energy Conference (ENERGYCON 2022)*, 1–5. IEEE.
- Tafazoli, S.; and Keogh, E. 2023. Matrix Profile XXVIII: Discovering Multi-Dimensional Time Series Anomalies with K of N Anomaly Detection. In *Proceedings of the 2023 SIAM International Conference on Data Mining (SDM 2023)*, 685–693. SIAM.

Torkamani, S.; and Lohweg, V. 2017. Survey on time series motif discovery. *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery*, 7(2): e1199.

Yeh, C.-C. M.; Kavantzias, N.; and Keogh, E. 2017. Matrix profile VI: Meaningful multidimensional motif discovery. In *2017 IEEE International Conference on Data Mining (ICDM 2017)*, 565–574. IEEE.

Yeh, C.-C. M.; Zhu, Y.; Ulanova, L.; Begum, N.; Ding, Y.; Dau, H. A.; Silva, D. F.; Mueen, A.; and Keogh, E. 2016. Matrix profile I: all pairs similarity joins for time series: a unifying view that includes motifs, discords and shapelets. In *2016 IEEE 16th International Conference on Data Mining (ICDM 2016)*, 1317–1322. IEEE.

Zhang, J.; Zhang, X.; Hong, D.; Gupta, R. K.; and Shang, J. 2025. Contextual inference from sparse shopping transactions based on motif patterns. *IEEE Transactions on Knowledge and Data Engineering (TKDE)*, 37(2): 572–583.

Zhong, S.; and Mueen, A. 2024. MASS: distance profile of a query over a time series. *Data Mining and Knowledge Discovery (DMKD)*, 38(3): 1466–1492.

Zhou, J.; Cui, G.; Hu, S.; Zhang, Z.; Yang, C.; Liu, Z.; Wang, L.; Li, C.; and Sun, M. 2020. Graph neural networks: A review of methods and applications. *AI open*, 1: 57–81.

Zhu, Y.; Mueen, A.; and Keogh, E. 2019. Matrix profile IX: Admissible time series motif discovery with missing data. *IEEE Transactions on Knowledge and Data Engineering (TKDE)*, 33(6): 2616–2626.

Zhu, Y.; Yeh, C.-C. M.; Zimmerman, Z.; Kamgar, K.; and Keogh, E. 2018. Matrix profile XI: SCRIMP++: time series motif discovery at interactive speeds. In *2018 IEEE International Conference on Data Mining (ICDM 2018)*, 837–846. IEEE.

Zhu, Y.; Zimmerman, Z.; Senobari, N. S.; Yeh, C.-C. M.; Funning, G.; Mueen, A.; Brisk, P.; and Keogh, E. 2016. Matrix profile ii: Exploiting a novel algorithm and gpus to break the one hundred million barrier for time series motifs and joins. In *2016 IEEE 16th International Conference on Data Mining (ICDM 2016)*, 739–748. IEEE.

Zimmerman, Z.; Senobari, N. S.; Funning, G.; Papalexakis, E.; Oymak, S.; Brisk, P.; and Keogh, E. 2019. Matrix profile XVIII: time series mining in the face of fast moving streams using a learned approximate matrix profile. In *2019 IEEE International Conference on Data Mining (ICDM 2019)*, 936–945. IEEE.