MLAT: Metric Learning for kNN in Streaming Time Series

Dongmin Park, Susik Yoon, Hwanjun Song, Jae-Gil Lee*
Graduate School of Knowledge Service Engineering, KAIST
{dongminpark,susikyoon,songhwanjun,jaegil}@kaist.ac.kr

ABSTRACT

Learning a good distance measure for distance-based classification in time series leads to significant performance improvement in many tasks. Specifically, it is critical to effectively deal with variations and temporal dependencies in time series. However, existing metric learning approaches focus on tackling variations mainly using a strict alignment of two sequences, thereby being not able to capture temporal dependencies. To overcome this limitation, we propose MLAT, which covers both alignment and temporal dependencies at the same time. MLAT achieves the alignment effect as well as preserves temporal dependencies by augmenting a given time series using a sliding window. Furthermore, MLAT employs time-invariant metric learning to derive the most appropriate distance measure from the augmented samples which can also capture the temporal dependencies among them. We show that MLAT outperforms other existing algorithms in the extensive experiments on various real-world data sets.

CCS CONCEPTS
• Mathematics of computing → Time series analysis; • Theory of computation → Machine learning theory; • Computing methodologies → Supervised learning by classification.

KEYWORDS
Streaming Time Series, Metric Learning, Optimization

1 INTRODUCTION

Streaming time series classifications play an increasingly important role in activity recognition [15] and fraud detection [12]. However, since the number of labels in streaming time series data is often insufficient to build a high-quality classifier [22], k-nearest neighbor (kNN), a non-parametric method, is widely used and it empirically results in high accuracy in several applications [6].

In accordance with this trend, many studies have been focused on improving the performance of kNN by designing more appropriate distance measures for a given data set. Recently, various time series metric learning approaches [3, 14, 20] have been developed to achieve this goal. Because variations in time series data, such as sequence shifting and scaling, is one of the main challenges, the existing algorithms consist of the two phases: (i) alignment and (ii) metric learning. As shown in Figure 1(a), the subsequences of the time series data are aligned in pairs to match the optimal time steps (dotted line), rendering them robust to variation. Metric learning is subsequently conducted on the local distance \( D_l \) computed from the matched time steps, which minimizes the distance of pairs with the same label and maximizes for pairs with different labels.

However, since the local distance only considers the difference between features at a single matched time step, metric learning on the local distance does not capture temporal dependencies appearing across consecutive time steps. Temporal dependency is known to greatly enhance the performance of time series classifications [10]. For example, when we try to distinguish between forwards and backwards from a set of motion images of a walking person, an image of a single time step gives very limited information on the direction, whereas a set of images of consecutive time steps provides clear clues. It is, therefore, more effective to distinguish samples with different labels by applying metric learning on the window distance \( D_w \) computed from consecutive time steps (e.g., \( \{t_1, t_2, t_3\} \)), as shown in Figure 1(b). The existing algorithms miss out this opportunity as they focus on alignment at the expense of the temporal dependencies. This calls for a new method that can achieve both alignment and temporal dependencies.

In this paper, we propose a novel metric learning algorithm for kNN in streaming time series, called MLAT (Metric Learning considering Alignment and Temporal dependencies). To account for both alignment and temporal dependency in learning a distance measure, MLAT consists of the two main phases:
1. **Sliding Window Augmentation**: MLAT augments a given time series using sliding window sampling [24] to achieve the alignment effect while preserving the temporal dependencies. As shown in Figure 2, by extracting all possible samples as sets of consecutive time steps with a fixed length, the samples having similar patterns can be well-aligned even without explicit time step matching. At the same time, because consecutive time steps are well preserved in a sample, MLAT can exploit the temporal dependencies between them.

2. **Time-Invariant Metric Learning**: The temporal dependencies between features of augmented samples are *time-invariant*, meaning that the dependency of two features in a fixed time difference is consistent regardless of their absolute temporal locations; i.e., the dependency of two features at time \( t \) and \( t+1 \) is equal to that of the two features at time \( t+1 \) and \( t+2 \). Thus, MLAT tries to fulfill this inherent characteristic in learning distance metric using the large margin nearest neighbor (LMNN) [13], constraining a Mahalanobis matrix to be a *Block Toeplitz* [8] structure.

Extensive experiments on four real-world streaming time series data sets indicate that MLAT results in promising kNN performance.

## 2 RELATED WORK

We briefly review several existing studies to find better distance measures in time series data with respect to the following aspects: (i) alignment to appropriately handle variation, and (ii) metric learning to learn a better distance function.

### 2.1 Alignment

Dynamic time warping (DTW) [1] is the most representative method to match the time steps of two samples in order to find the best warping path. Numerous variants of DTW have been studied, whose focus is on resolving the inefficiencies in time step matching [15, 17] and the invalidity of the triangle inequality in the learned distance [4, 11]. However, DTW-based alignment methods fail to deal with the temporal dependencies because they explicitly match the time steps.

### 2.2 Metric Learning

Most time series metric learning algorithms in the literature apply metric learning to the local distance computed using DTW based alignment. LDML-DTW [14] and LMNN-DTW [20] first match time steps by multivariate dynamic time warping (MDTW) [1], and subsequently learn the Mahalanobis matrix of the local distance using LogDet divergence and Large Margin Nearest Neighbor (LMNN) [23], respectively. DECADE [3] devised a new alignment method to learn the valid distance measure while using deep networks to capture the complex dependencies in the local distance. There have been a few studies for learning the distance representation in the embedded space obtained from the last hidden layer of LSTM [16].

However, to the best of our knowledge, no studies yet simultaneously consider alignment and temporal dependencies in time series metric learning.

## 3 PRELIMINARY

### 3.1 Problem Setting

We introduce the main concepts in the kNN classification problem on streaming time series in the following definitions, as also illustrated in Figure 3.

**Definition 3.1.** *(Streaming time series)* A multivariate streaming time series \( x = [x_1, x_2, \ldots, x_T] \) is a sequential observation of \( x_i \in \mathbb{R}^d \) with \( d \) features, where \( T \) is the length of \( x \).

**Definition 3.2.** *(State)* A state \( s_j \in [1, 2, \ldots, S] \) denotes the label of a sequence in a time series \( x \) during the time period \( T_j \subset \mathbb{T} \), where \( S \) is the number of states in \( x \) and \( j \) is the index of the state. The time series \( x \) consists of various sizes of *walks* with different states; for example, a time series \( x \) collected from wearable devices for 15 min consists of the two sequences: a "walk" state of the first 10 min and a "run" state of the last 5 min.

**Definition 3.3.** *(Sample)* A sample \( X_t = [x_t, \ldots, x_{t+w-1}] \in \mathbb{R}^{d \times w} \) is a subsequence of size \( w \) extracted from a certain sequence in time period \( T_j \) of a time series \( x \). It consists of the consecutive time steps from \( t \) to \( t + w - 1 \). The state \( s_j \) of the sequence becomes the label \( y_t \) of \( X_t \).

**Definition 3.4.** *(kNN classification problem)* Let \( \mathbb{D} \) be a set of samples from a time series \( x \). The kNN classification problem in streaming time series classifies the label of a sample \( X_{\text{new}} \) by referring to that of the k nearest neighbors of \( X_{\text{new}} \) in \( \mathbb{D} \).

### 3.2 Large Margin Nearest Neighbor [23]

Most metric learning methods aim at learning a Mahalanobis distance matrix \( M \in \mathbb{R}^{d \times d} \), where the distance is defined as:

\[
D_M(X_i, X_j) = (X_i - X_j)^T M (X_i - X_j),
\]

where \( X_i \) and \( X_j \in \mathbb{R}^{d \times w} \) are two samples and \( M \in \mathbb{R}^{d \times w \times d \times w} \) is a positive semidefinite matrix. Given a set of samples \( \mathbb{D} \), for each sample \( X_i \in \mathbb{D} \), LMNN first finds the \( k \) nearest neighbors of \( X_i \) based on the Euclidean distance and calls them target neighbors. Subsequently, LMNN learns the optimal Mahalanobis distance matrix \( M \) which minimizes the following loss function:

\[
\min_{M \succeq 0} \sum_{i,j} \eta_{ij}D_M(X_i, X_j) + c \sum_{i,j} (1 - y_{ij}) \left[ 1 + D_M(X_i, X_j) + D_M(X_i, X_j) \right],
\]

where \( c \in [0, 1] \) controls the weights of the two penalizing terms. \( \eta_{ij} \in [0, 1] \) indicates whether \( X_i \) is a target neighbor of \( X_j \), \( y_{ij} \in \{0, 1\} \) indicates whether \( X_i \) and \( X_j \) have the same label or not.
4 METHODOLOGY

MLAT consists of two phases: (1) Sliding window augmentation, and (2) Time-invariant metric learning. Algorithm 1 describes the overall procedure of MLAT.

4.1 Phase I: Sliding Window Augmentation

Phase I enables metric learning to consider alignment and temporal dependencies concurrently. Specifically, all possible samples in $x$ can be extracted by sliding a fixed sized window from the beginning of $x$. Some parts of $x$ where the state switches are exempted from the augmentation. These augmented samples become the training set for metric learning and help to achieve the alignment effect. As in Figure 4(a), a sample $X$ has the most well-aligned training sample $X_{\text{target}}(i)$ as its neighbor with euclidean distance. Therefore, MLAT, which is based on LMNN, exploits the well-aligned samples $X_{\text{target}}(i)$ as the target neighbor of $X$ if they share the same label and (i) temporal dependencies concurrently. Specifically, all possible samples in $x$.

4.2 Phase II: Time-Invariant Metric Learning

Phase II of the augmented samples is mostly overlapped. This leads to the covariance matrix of augmented samples in the form of Block Toeplitz [8] as stated in Definition 4.1, which is proven by Theorem 4.2.

**Definition 4.1.** (Block Toeplitz [8]) A $dw \times dw$ block toeplitz matrix $A$ with $d \times d$ sub-blocks $A^{(m)}$ has the following form:

$$A = \begin{bmatrix} A^{(0)} & A^{(1)} & \cdots & A^{(w-1)} \\ A^{(1)} & A^{(0)} & \cdots & A^{(w-2)} \\ \vdots & \vdots & \ddots & \vdots \\ A^{(w-1)} & A^{(w-2)} & \cdots & A^{(0)} \end{bmatrix},$$

(3)

where the sub-block $A^{(m)}$ appears $w - m$ times in $A$ and has the same value at all occurrences.

**Theorem 4.2.** Let $D_{\alpha} = \{X_1, X_2, \ldots, X_L\} \subseteq \mathbb{R}^{d \times L}$ be a set of samples obtained by sliding window augmentation, where $X_i \in \mathbb{R}^d$ and $L \gg w$. The covariance matrix $\Sigma \in \mathbb{R}^{d \times dw \times dw}$ of $D_{\alpha}$ is in the form of Block Toeplitz.

**Proof.** $D_{\alpha}$ is represented with $w$ row partitions as follows:

$$D_{\alpha} = \begin{bmatrix} x_1 & x_2 & x_3 & \cdots & x_{L-w+1} \\ x_2 & x_3 & x_4 & \cdots & x_{L-w+2} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ x_w & x_{w+1} & x_{w+2} & \cdots & x_L \end{bmatrix},$$

(4)

where $x_i \in \mathbb{R}^d$. Since most portions in each row partition overlap, the mean $\mu_p \in \mathbb{R}^d$ of each row partition is approximately the same. Then, by the definition of the covariance matrix, $\Sigma = 1/L \cdot (D_{\alpha} - \mu)(D_{\alpha} - \mu)^T$ approximately takes Block Toeplitz form, where $\mu = [\mu_1, \cdots, \mu_p]^T$, $1_L^T \in \mathbb{R}^{dw \times L}$.

The $(i, j)$-th element of the covariance matrix corresponds to the dependency between the $i$-th and $j$-th features, and if the matrix is Block Toeplitz, the dependency between the two features is time-invariant [9]. Note that the time-invariant dependencies of the augmented samples should be preserved when learning distance.

4.2.2 Time-Invariant Constraints. Here, we first analyze how the Mahalanobis matrix handles the dependency between two features. Eq. (1) can be represented as follows:

$$D_M(X_i, X_j) = tr(M(X_i - X_j)^T(X_i - X_j))$$

$$= \sum_{ij} M_{ij}(X_{i}^{(i)} - X_{j}^{(i)})(X_{i}^{(j)} - X_{j}^{(j)}),$$

(5)

where $X_{i}^{(i)}$, $X_{j}^{(i)}$ are the $i$-th feature scale value of a sample $X_i$, and $M_{ij}$ is the $(i, j)$-th element of $M$. $M_{ij}$ decides how much the product of the distance between $i$-th features and that of $j$-th features affects the total distance, meaning that $M_{ij}$ handles the dependency between the $i$-th and $j$-th features.

To preserve the time-invariant dependencies, we propose the following time-invariant metric learning framework:

$$\min_{M} \mathcal{E}^{\text{lmnn}}(M)$$

subject to: $M \succeq 0$, $M \in \mathcal{T}$,

(6)

where $\mathcal{T}$ is a set of $dw \times dw$ symmetric Block Toeplitz matrices. By constraining $M$ to be Block Toeplitz, the distances between two
features in the same time difference should be regarded equally, with the same values as in $M$. Thus, the optimal matrix can effectively capture the time-invariant dependencies structure.

4.2.3 Optimization. We solve the proposed optimization problem using the alternating direction method of multipliers (ADMM) [2]. We transform Eq. (6) into an ADMM-friendly form as follows:

$$
\mathcal{L}_\rho(M, Z, U) = E^{l_{max}}(M) + \frac{\rho}{2} \| M - Z + U \|_F^2.
$$

subject to $Z \in T$, $Z = M, M \succeq 0$.

Then, the following three steps are repeated until convergence.

(a) $M^{k+1} = \arg \min_M \mathcal{L}_\rho(M, Z^k, U^k)$

(b) $Z^{k+1} = \arg \min_Z \mathcal{L}_\rho(M^{k+1}, Z, U^k)$

(c) $U^{k+1} = U^k + (M^{k+1} - Z^{k+1})$.

M-update: The M-update can be written as:

$$
M^{k+1} = \arg \min_M E^{l_{max}}(M) + \frac{\rho}{2} \| M - Z^k + U^k \|_F^2.
$$

This problem can be solved by the gradient descent method. To ensure a positive semi-definite $M$, MLAT factorizes $M$ as $M = L^T L$ and updates $L$ through sub-gradient descent. Here, the sub-gradient of Eq. (9) with respect to $L$ is $\frac{\partial E^{l_{max}}}{\partial L} = 2L^T(\frac{\partial E^{l_{max}}}{\partial M} + M - Z^k) + U^k$ by the chain rule.

Z-update: The closed form solution of the Z-update is as follows:

$$
Z^{k+1}_{i,j} = \frac{R^{(m)}_{i,j,l}}{R^{(m)}_{i,j,l}} \sum_{l=1}^{K^{(m)}} (M^{k+1} + U^k)_{i,j,l}^{(m)} / R^{(m)}_{i,j,l},
$$

where $R^{(m)}_{i,j,l}$ is the index $(x,y)$ at $Z$ of the $(i,j)$-th element of the $l$-th occurrence sub-matrix $Z^{(m)}$, and $R^{(m)}_{i,j,l}$ is the number of occurrences of $Z^{(m)}$ in $Z$. Refer to [9] for details.

5 EXPERIMENTS

To validate the superiority of MLAT, we performed the kNN classification task on four real-world streaming time series data sets. The experimental results confirmed that MLAT maintains its dominance over other algorithms.

5.1 Experiment Setup

5.1.1 Data Sets. The statistics of the four benchmark data sets are summarized in Table 1. Note that we only used 5 ambiguous labels defined as normal activities in EMG and Vicon data sets for a more difficult kNN task.

5.1.2 Algorithms. We compared MLAT with four existing algorithms for measuring distance in time series:

- **ED**: Euclidean distance (baseline).
- **MDTW** [1]: Multivariate dynamic time warping.
- **LDMLT** [14]: Learning local distance by DTW alignment.
- **LMNN** [13]: Large Margin Nearest Neighbor.
- **MLAT**: Our proposed method.

5.2 Experimental Results

5.2.1 Overall Accuracy. Table 2 shows the kNN accuracy of the five algorithms. In the SCMA, EMG, and Vicon data sets, MLAT achieved the highest accuracy compared with other algorithms. LDMLT produced the highest accuracy for the AReM data set, but MLAT also achieved a significant improvement compared with the remaining algorithms. This emphasizes the need to consider both alignment and temporal dependencies.

5.2.2 Effect of Phases I and II. All the existing algorithms were also evaluated on augmented samples obtained in Phase I. As shown in Table 3, the accuracy of the algorithms were significantly improved by up to 3.3% compared with the accuracy from randomly extracted samples. This shows that augmenting well-aligned samples is evidently beneficial for the time series classification.

In addition, MLAT still outperformed LMNN by up to 9.8%, though LMNN performed on augmented samples. The performance of metric learning for time series can thus be further boosted by preserving the time-invariant characteristic.

6 CONCLUSION

In this paper, we proposed MLAT, a novel metric learning algorithm that considers the two main characteristics in time series, i.e., variation and temporal dependencies, by using sliding window augmentation and time-invariant metric learning, respectively. Using four real-world data sets, we showed that MLAT outperforms the existing algorithms in most cases. For future work, we will tackle more challenging settings where more complex variations and temporal dependencies exist.
ACKNOWLEDGMENTS
This work was partly supported by the National Research Foundation of Korea (NRF) grant funded by the Korea government (Ministry of Science and ICT) (No. 2017R1E1A1A01075927) and the MOLIT (The Ministry of Land, Infrastructure and Transport), Korea, under the national spatial information research program supervised by the KAIA (Korea Agency for Infrastructure Technology Advancement) (19NSIP-B081011-06).

REFERENCES