Learning Discriminative Virtual Sequences for Time Series Classification

Abhilash Dorle¹, Fangyu Li², Wenzhan Song¹, Sheng Li¹
¹University of Georgia, Athens, Georgia, USA
²Kennesaw State University, Marietta, Georgia, USA
{AbhilashYeshwant.Dorle,wsong,sheng.lij}@uga.edu,ll6@kennesaw.edu

ABSTRACT
Temporal data are continuously collected in a wide range of domains. The increasing availability of such data has led to significant developments of time series analysis. Time series classification, as an essential task in time series analysis, aims to assign a set of temporal sequences to different categories. Among various approaches for time series classification, the distance metric learning based ones, such as the virtual sequence metric learning (VSLM), have attracted increased attention due to their remarkable performance. In VSLM, virtual sequences attract samples from different classes to facilitate time series classification. However, the existing VSLM methods simply employ fixed virtual sequences, which might not be optimal for the subsequent classification tasks. To address this issue, in this paper, we propose a novel time series classification method named Discriminative Virtual Sequence Learning (DVSL). Following the unified framework of sequence metric learning, our DVSL method jointly learns a set of discriminative virtual sequences that help separate time series samples in a feature space, and optimizes the temporal alignment by dynamic time warping. Extensive experiments on 15 UCR time series datasets demonstrate the efficiency of DVSL, compared with several representative baselines.

CCS CONCEPTS
\- Computing methodologies → Machine learning algorithms;
\- Information systems → Data mining.

KEYWORDS
Time Series Classification, Metric Learning, Virtual Sequences

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than ACM must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from permissions@acm.org.

Copyright © 2020 Association for Computing Machinery.
ACM ISBN 978-1-4503-6859-9/20/10 \$15.00
https://doi.org/10.1145/3340531.3412099

1 INTRODUCTION
Time series data are prevalent across many domains, such as health, finance, and entertainment. To name a few examples, data readings from IoT devices could help monitor the status of electrical systems; body wearable devices can provide insightful information about human activities and behavior; surveillance cameras deployed in transit systems could help address public security concerns. The increasing availability of time series data has largely driven the research efforts on time series analysis in recent years. Some fundamental time series analysis tasks include classification, prediction, anomaly detection, clustering and visualization [7–9]. Time series classification, as the primary task in this area, aims to separate temporal sequences into some predefined categories. A large number of time series classification methods have been proposed in the past decade, which can be roughly categorized as the local feature methods, deep learning based methods, and metric learning based methods. Local feature based methods extract representative attributes from time series, such as the shapelets, which are defined as time series subsequences that maximally represent a class [12, 15]. Deep learning based methods learn a latent feature space for time series data, which have obtained impressive performance [3, 4]. Metric learning based methods try to develop metrics that bring closer time series samples from the same class and meanwhile separate out those from different classes. A commonly used metric for time series classification is the dynamic time warping (DTW), which aligns time series through dynamic programming [1]. Many variants of DTW have been proposed to improve the performance and reduce the time complexity [16, 17].

Most recently, a virtual sequence metric learning (VSLM) framework for time series classification is proposed in [13]. This framework employs a set of virtual sequences that are inspired by the idea of virtual points [11]. In this framework, virtual sequences are predefined in the sequential data space, and the objectives of metric learning and temporal alignment can be jointly optimized. In particular, this framework brings samples from each class closer to a class-specific virtual sequence, such that the time series samples from different classes can be separated. Although this method has obtained quite promising results on time series classification [13], it still has some limitations. First, the performance of this method heavily relies on the quality of predefined virtual sequences. Since virtual sequence construction and time series classification are two isolated steps, such virtual sequences may not be optimal for the subsequent classification task. Second, the design of virtual sequences is very subjective, which may not fit various downstream applications in practice.

To overcome the limitations in existing work, in this paper, we propose a novel time series classification method by learning discriminative virtual sequences. Our method adaptively learns a set of virtual sequences for time series data, which can be seamlessly integrated with the unified sequence metric learning framework.
warping (DTW) is a commonly used method [1]. DTW based nearest neighbor has produced exceptional results in time series classification. However, the time complexity of DTW is very high. Several algorithms have been proposed to mitigate the complexity brought by DTW. In [16], some learned features are used to weight time series samples, leading to a weighted DTW method. This method involves representing the scarce training data in an embedded space that is then used for classification. In addition, a recent study takes into consideration the local shape of the samples to improve DTW [17]. As the similarity of local shapes matches, the proposed feature encoding produces better results.

Our method is closely related to the virtual sequence metric learning methods [13]. The concept of virtual point in metric learning was first proposed in [11]. Virtual points are predefined data points that can be used to assist classification in the metric space, by bringing closer examples of the same class to a particular virtual point. This method largely reduces the number of constraints in traditional metric learning. In [13], the concept of virtual points is extended to virtual sequences, which can be used to help sequence data classification. The training time series samples are brought closer to an associated virtual sequence, which generates small values for samples from the same class than those from different classes. A unified virtual sequence metric learning framework is also introduced in [13], which jointly learns the ground metric and aligns the time series samples with virtual sequences. However, the virtual sequences in [13] are predefined and also fixed during model training. The length of the virtual sequence is set to 1 ensuring that the alignment between sequences is unique. As a result, the model performance highly depends on the construction of virtual sequences. Different from existing work, our method adaptively learns a set of discriminative virtual sequences for time series classification, which could be easily adapted to time series data in different domains.

## 3 OUR APPROACH

### 3.1 Problem Statement

In this section we provide the definitions of time series, virtual sequence and time series classification.

**Definition 3.1. Time Series.** A time series \( X^n = [x_1, x_2, \ldots, x_L] \) is an ordered sequence of data, where \( x_t \) is the value of the time series at time stamp \( t \), and \( L \) is the length of the time series. The label of the time series \( X^n \) is denoted as \( y^n \).

**Definition 3.2. Virtual Sequence** [13]. A virtual sequence \( V^n = [v_1, \ldots, v_N] \in \mathbb{R}^{b \times d} \) is defined as a function of \( X^n \) and \( y^n \), i.e., \( V^n = f(X^n, y^n) \).

In [13], virtual sequences are used to assist sequence metric learning and sequence data classification.

**Definition 3.3. Time Series Classification.** Given a test time series \( X^t \), the goal of time series classification is to predict the corresponding class label \( y^t \).

### 3.2 Formulation

In this paper, we propose a novel time series classification method based on Discriminative Virtual Sequences Learning (DVSL). Let \( X = \{X^n, y^n\}_{n=1}^N \) denote a set of \( N \) time series samples. We assume that the time series belong to \( C \) different classes. In particular, a time series sample can be represented as \( X^n = [x^n_1, \ldots, x^n_L] \), where \( L \) is the length of \( X^n \). A Regressive Virtual Sequence Metric Learning (RVSML) framework is proposed in [13], which jointly optimizes a
Learning Discriminative Virtual Sequences for Time Series Classification

CIKM ’20, October 19–23, 2020, Virtual Event, Ireland

The major limitation of RVSML is that the virtual sequences are fixed during metric learning, which motivates us to explore adaptive and discriminative virtual sequences for time series classification. By considering virtual sequences \( V \) as variables, we formulate our method as:

\[
\min_{W, T, V} \Phi(X, D(W), T, V) + \Theta(V) + \Omega(T),
\]

where \( W \) is the parameter matrix and \( \Theta(V) \) is a regularization term.

3.2.1 Distance Metric Learning. The first term of Eq. (2) aims to learn a ground metric \( D(W) \) that optimizes the distance between the virtual sequence and the training time series samples, given the alignment matrix \( T \). By using the virtual sequences, DVSL brings closer time series samples from the same class to a specific virtual sequence, and meanwhile push away samples from different classes. Specifically, \( \Phi(X, W, T, V) \) is formulated as:

\[
\Phi(X, D(W), T, V) = \frac{1}{N} \sum_{n=1}^{N} \Phi^n(D^n(W)) + \lambda \| W \|_F^2
\]

\[
= \frac{1}{N} \sum_{n=1}^{N} \sum_{l=1}^{l_n} \sum_{j=1}^{l_n} \sum_{i=1}^{l_n} \| X^n_i - V^n_i - T^n_j l_i \|_F^2 + \lambda \| W \|_F^2
\]

where \( \lambda \) is a trade-off parameter, and \( l_n \) is the length of virtual sequence. \( T^n_j l_i \) is an element in the alignment matrix \( T \), which captures how closely the training sample and the virtual sequence align.

3.2.2 Virtual Sequence Learning. As discussed above, the virtual sequences used in RVSML [13] are fixed, which may not be optimal for the subsequent time series classification task. Instead, our DVSL method aims to learn a set of discriminative virtual sequences that can directly benefit the classification task. In particular, the discriminative virtual sequences shall be well separated in the data space. To this end, we design the second term in Eq. (2) as follows:

\[
\Theta(V) = \frac{1}{N} \sum_{n=1}^{N} \sum_{c=1}^{c_n} \sum_{i=1}^{l_n} \| V^n_i - V^n_i \|_2^2
\]

where \( V_j \) and \( V_i \) denote virtual sequences. With this term, we push the virtual sequences far away from each other by maximizing their pairwise distances in the sequential data space. To align with the minimization problem in Eq. (2), a negative sign in added to Eq. (4).

3.2.3 Objective Function. Combining the distance metric learning, virtual sequence learning and the regularization term, the overall objective function of our DVSL method is written as:

\[
\min_{W, T, V} \mathcal{L} = \sum_{n=1}^{N} \sum_{c=1}^{c_n} \sum_{i=1}^{l_n} \frac{1}{N} \sum_{j=1}^{l_n} \| T^n_j l_i X^n_i - V^n_i \|_2^2 + \lambda \| W \|_F^2
\]

\[
- \sum_{n=1}^{N} \sum_{c=1}^{c_n} \sum_{i=1}^{l_n} \sum_{j=1}^{l_n} \| V^n_i - V^n_i \|_2^2 + \Omega(T).
\]

Table 1: Times series classification results of the proposed DVSL method and baselines on UCR datasets. The names of the datasets have been shortened to accommodate the details

<table>
<thead>
<tr>
<th>Dataset</th>
<th>ED</th>
<th>DTW</th>
<th>LSDTW</th>
<th>RVSML</th>
<th>DVSL</th>
</tr>
</thead>
<tbody>
<tr>
<td>ArrowHead</td>
<td>80.00</td>
<td>70.29</td>
<td>73.14</td>
<td>74.86</td>
<td>72.00</td>
</tr>
<tr>
<td>Beef</td>
<td>66.67</td>
<td>63.33</td>
<td>83.33</td>
<td>83.33</td>
<td>90.00</td>
</tr>
<tr>
<td>Car</td>
<td>73.33</td>
<td>73.33</td>
<td>86.67</td>
<td>80.00</td>
<td>83.50</td>
</tr>
<tr>
<td>ChlConcent</td>
<td>65.00</td>
<td>64.84</td>
<td>72.11</td>
<td>60.83</td>
<td>77.43</td>
</tr>
<tr>
<td>Coffee</td>
<td>100.00</td>
<td>100.00</td>
<td>100.00</td>
<td>100.00</td>
<td>100.00</td>
</tr>
<tr>
<td>ECG200</td>
<td>88.00</td>
<td>77.00</td>
<td>85.00</td>
<td>85.00</td>
<td>83.50</td>
</tr>
<tr>
<td>ECGFiveDays</td>
<td>79.67</td>
<td>76.77</td>
<td>90.01</td>
<td>95.00</td>
<td>97.36</td>
</tr>
<tr>
<td>Herring</td>
<td>51.56</td>
<td>53.13</td>
<td>46.88</td>
<td>65.60</td>
<td>65.63</td>
</tr>
<tr>
<td>InsectWingb</td>
<td>56.16</td>
<td>35.51</td>
<td>50.91</td>
<td>58.83</td>
<td>58.19</td>
</tr>
<tr>
<td>Meat</td>
<td>93.33</td>
<td>93.33</td>
<td>80.00</td>
<td>90.00</td>
<td>98.83</td>
</tr>
<tr>
<td>MPhaOLAge</td>
<td>51.95</td>
<td>50.00</td>
<td>56.49</td>
<td>48.70</td>
<td>58.18</td>
</tr>
<tr>
<td>OliveOil</td>
<td>86.67</td>
<td>83.33</td>
<td>83.33</td>
<td>73.30</td>
<td>84.67</td>
</tr>
<tr>
<td>SonyAlbR1</td>
<td>69.55</td>
<td>72.55</td>
<td>78.20</td>
<td>83.86</td>
<td>76.16</td>
</tr>
<tr>
<td>TwoLeadECG</td>
<td>74.71</td>
<td>90.52</td>
<td>93.15</td>
<td>91.83</td>
<td>91.60</td>
</tr>
<tr>
<td>Wine</td>
<td>61.11</td>
<td>57.41</td>
<td>57.41</td>
<td>59.25</td>
<td>65.00</td>
</tr>
</tbody>
</table>

3.3 Optimization

Although the objective function in Eq. 5 is not jointly convex with respect to all the variables \( W, V, T \) it is convex to each variable separately when the others are fixed. Thus, we alternatively update these variables. In particular, for the subproblem w.r.t. \( T \), we implement it with DTW that is solved by dynamic programming. For \( W \) and \( V \), we employ a gradient descent approach by initializing the variables \( W^{(0)} \) and \( V^{(0)} \) with random values and then updating them with the following rules:

\[
W^{(t+1)} = W^{(t)} - \gamma \frac{\partial L}{\partial W},
\]

\[
V^{(t+1)} = V^{(t)} - \gamma \frac{\partial L}{\partial V},
\]

where \( \gamma \) is a learning rate. The detailed derivatives with respect to \( W \) and \( V \) are written as:

\[
\frac{\partial L}{\partial W} = \sum_{n=1}^{N} \sum_{c=1}^{c_n} \sum_{i=1}^{l_n} \sum_{j=1}^{l_n} t^n_{ij} x^n_i x^n_j + \lambda N.
\]

\[
\frac{\partial L}{\partial V} = \sum_{n=1}^{N} \sum_{c=1}^{c_n} \sum_{i=1}^{l_n} \sum_{j=1}^{l_n} t^n_{ij} x^n_i x^n_j + \lambda N,
\]

where \( C = \sum_{n=1}^{N} \sum_{i=1}^{l_n} t^n_{ij} N - \sum_{n=1}^{N} \sum_{i=1}^{l_n} V^n i) / C, \) where \( C = \sum_{n=1}^{N} \sum_{i=1}^{l_n} t^n_{ij} x^n_i x^n_j + \lambda N. \)

After optimizing the virtual sequences \( V \) and the ground metric \( D(W) \), we employ the one nearest neighbor (1-NN) classifier with DTW distance measure for time series classification.

4 EXPERIMENTS

In this section, we evaluate the performance of our DVSL method and compare it with baseline methods on benchmark datasets.

Datasets. We evaluate our method thoroughly on the UCR time series archive [2]. The repository contains datasets from various real world domains. It consists of time series data from various sources like sensor data, image data and spectrograph data. We perform experiments on 15 datasets to show the effectiveness of our method.
5 CONCLUSION

In this paper, we proposed a novel time series classification method based on Discriminative Virtual Sequences Learning (DVSL). The goal of DVSL is to bring closer time series samples from the same class to a specific virtual sequence and meanwhile push away samples from different classes. Different from existing work, DVSL adaptively learns a set of discriminative virtual sequences. A new objective function is formulated and a gradient descent algorithm is designed for optimization. Experiments on 15 UCR time series datasets demonstrate that our method outperforms several representative baselines.

REFERENCES