Engineering Applications of Artificial Intelligence 116 (2022) 105331

Contents lists available at ScienceDirect

Engineering Applications of Artificial Intelligence

journal homepage: www.elsevier.com/locate/engappai

Semi-supervised Time Series Classification Model with Self-supervised Learning

Liang Xi [a,*], Zichao Yun [a], Han Liu [a], Ruidong Wang [a], Xunhua Huang [a], Haoyi Fan [b]

[a] Harbin University of Science and Technology, 52 Xuefu Road, Harbin, 150080, China
[b] Zhengzhou University, 100 Kexuedadao Road, Zhengzhou, 450001, China

A R T I C L E   I N F O

Dataset link: https://github.com/mrxiliang/sstsc.git

Keywords:
Time series classification
Semi-supervised learning
Self-supervised learning
Temporal relation
Convolutional neural network

A B S T R A C T

Semi-supervised learning is a powerful machine learning method. It can be used for model training when only part of the data are labeled. Unlike discrete data, time series data generally have some temporal relation, which can be considered as a supervised signal in semi-supervised learning to supervise the learning of unlabeled time series data. However, the currently known semi-supervised time series classification (TSC) methods always ignore or under-explore the temporal relation structure and fail to fully use the unlabeled time series data. Therefore, we propose a Semi-supervised Time Series Classification Model with Self-supervised Learning (SSTSC). It takes self-supervised learning as the auxiliary task and jointly optimizes it with the main TSC task. Specifically, it performs the TSC task on the labeled time series data; For the unlabeled time series data, it splits the "past-anchor-future" segments and constructs the positive/negative temporal relation samples with different combinations to accurately predict the temporal relations and capture the higher-quality semantic context in self-supervised learning as a supervised signal for TSC task. Experimental results demonstrate that SSTSC has better effects than the baselines from different perspectives.

1. Introduction

As one of the important data types, time series data exists in various fields (Xing et al., 2012; Moon et al., 2021). Data are interdependent and have high temporal relation, which is the most significant difference between time series data and discrete data. Time Series Classification (TSC) is to distinguish time series data with specific common attributes or features. TSC can be seen in many application scenarios, such as ElectroCardioGraph (ECG) Diagnosis (Ribeiro et al., 2020), Satellite Image Detection (Verbesselt et al., 2010), Human Activities Recognition (Nweke et al., 2018), and Fault Diagnosis of Train Bearing (Li et al., 2021), as shown in Fig. 1. Large amounts of time series data are constantly changing over time. Moreover, there are a lot of noise data, which makes TSC a complicated task in data mining (Esling and Agon, 2012).

It has been widely proved that deep learning framework can learn advanced features in TSC (Chen et al., 2021; Ismail Fawaz et al., 2019). At present, supervised learning is mainly a mainstream method. For example, Multi-Layer Perceptron (MLP) learns the temporal relations directly through the fully connected neural network. Fully Convolutional Network (FCN), Residual Network (ResNet), and LSTM_FCN that combines Long Short Term Memory (LSTM) and FCN are also applied to feature learning of time series data (Wang et al., 2017; Karim et al., 2017). These supervised methods require a large amount of labeled data. However, obtaining so many labeled data is costly and impractical. Therefore, the current research focus has gradually shifted to semi-supervised and self-supervised learning.

The Semi-supervised learning methods mainly use the unlabeled data for training when only part of the data are labeled (Hailat et al., 2018), and have been widely applied to Video Indexing (Husain and Meena, 2019), Image Classification (Zhang et al., 2019), Anomaly Detection (Kumar and Awate, 2020) and other fields. Recently, semi-supervised learning with deep neural network (DNN) has been applied to the TSC. The existing ideas mainly adopt the pseudo-label strategy. It selects the class with the maximum predicted probability and uses it as the label. This strategy favors a low-density separation among classes, which is a common hypothesis for semi-supervised learning (Lee, 2013). However, it is vulnerable to noise, and the label may be mis-marked (Rizve et al., 2021). In order to avoid this problem, Π-Model forms a consensus prediction of the unlabeled data on different epochs under different regularization and input augmentation conditions (Laine and Aila, 2016). However, Π-Model ignores the temporal relation structure and cannot fully utilize the unlabeled data.

Self-supervised learning is one of the unsupervised learning methods, mainly using automatically self-generated labels to train Convolutional Neural Network (CNN) (Jing and Tian, 2020), aiming to capture more useful semantic context from the implicit structure of

* Corresponding author.
E-mail address: xiliang@hrbust.edu.cn (L. Xi).

https://doi.org/10.1016/j.engappai.2022.105331
Received 4 January 2022; Received in revised form 3 August 2022; Accepted 9 August 2022
Available online 1 September 2022

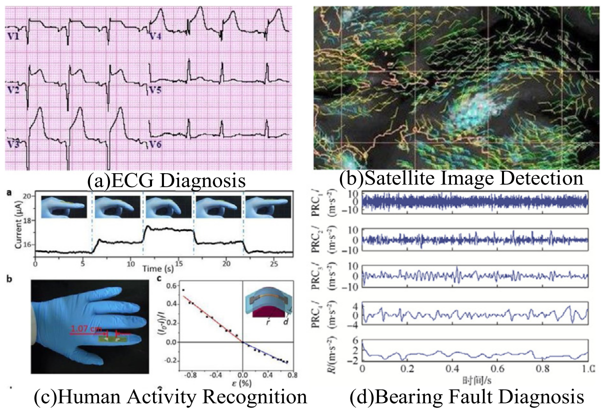

Fig. 1. Some TSC application scenarios.

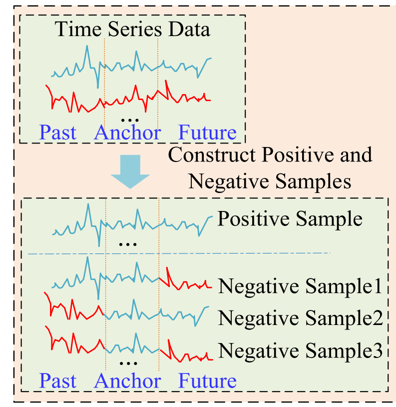

Fig. 2. Construction of positive/negative temporal relation samples.

unlabeled data (Fan et al., 2020; Jawed et al., 2020). But there are still some problems, such as insufficient use of unlabeled data and lack of considering the inhere temporal relation among time series. In our previous study, we proposed the SemiTime model to predict the temporal relations by sampling the "past-future" segments and constructing the positive/negative temporal relation samples, so as to get the useful semantic context of unlabeled time series (Fan et al., 2021). However, SemiTime only considers the temporal relation with the future (regards the "past" segment as the "anchor" segment), which is not enough to fully explore the semantic context of the temporal relation with the "past" and "future" together. Moreover, the type and quantity of negative temporal relation samples are not sufficient. But in fact, better feature context representations can be extracted by a large training set with richer negative temporal relation samples (He et al., 2020).

Therefore, we continue our research, proposing a **S**emi-supervised **T**ime **S**eries **C**lassification Model with **S**elf-supervised Learning (termed as SSTSC): for the labeled time series, SSTSC conducts the supervised classification directly; Meanwhile, SSTSC splits the unlabeled time series data into three segments to obtain the segment groups of "past", "anchor" and "future"; Then, through the different combinations shown in Fig. 2, we can obtain more types and quantities of negative temporal relation samples, reveal the temporal relations among the "past", "anchor" and "future" segments to further extract the semantic context, and use them as supervised signal to improve the performance of TSC task.

The main contributions of this paper can be summarized as follows:

(1) We propose a semi-supervised learning method for TSC, which uses a self-supervised learning module for the semantic context of temporal relation and utilizes it to assist the supervised module for TSC.

(2) For the self-supervised learning module, we design a temporal relation segments sampling and construction strategy with "past-anchor-future", that enables SSTSC to extract the higher-quality semantic context from the unlabeled time series data.

(3) We perform different experiments on eight datasets with different characteristics. Experimental results highlight that SSTSC outperforms the state-of-the-art.

The remainder of this paper is organized as follows. Section 2 reviews related TSC methods. The proposed model is described in Section 3. Section 4 presents our experimental results and analysis, while Section 5 concludes this paper.

2. Related work

2.1. TSC models

Most TSC has concentrated on similarity in time (Lines and Bagnall, 2015). Before deep learning became widely applied, Dynamic Time Warping (DTW), K Nearest Neighbor (KNN) algorithm, their combination and variants have been proven to be effective for TSC tasks (Bagnall et al., 2017; Keogh and Pazzani, 2001). In addition, there are also some works to perform TSC tasks by extracting features through time sub-series or interval partitioning (Ye and Keogh, 2011; Rodríguez et al., 2005; Large et al., 2019). Recently, deep learning frameworks have been applied to TSC tasks. In Wang et al. (2017), FCN and ResNet are used for TSC. Fawaz et al. (2018) propose a transfer learning framework to transfer deep CNNs for TSC. Zhao et al. (2017) propose a CNN-based model to perform TSC and temporal relation prediction tasks, respectively.

All the above models belong to supervised classification, which need the support of a large number of labeled data and are only applicable to specific data scenarios. In order to solve these problems, semi-supervised and self-supervised TSC models were proposed in some works.

2.2. Semi-supervised learning

Semi-supervised learning is a powerful strategy to mitigate the dependence on large amounts of labeled data by making full use of unlabeled data (Rasmus et al., 2015). It trains the labeled data with the help of unlabeled data, and makes up for the deficiency of labeled data. The semi-supervised classifier is better than the classifier that only trains labeled data.

The pseudo-label is one of the main strategies of semi-supervised learning (Rizve et al., 2021). The core is to use unlabeled data to generate pseudo-label for supervised training. Other existing semi-supervised models are mainly applied to image-related tasks, such as Π-Model (Laine and Aila, 2016) mentioned above and the similar Γ-Model (Rasmus et al., 2015). Π -Model performs unknown-label prediction for model training and gets ideal performance. In addition, some works (Berthelot et al., 2019; Pavllo et al., 2019) combine consistency regularization with pseudo-label. However, these methods have some disadvantages: they are not initially used for the TSC task, and ignore the temporal relation structure; it is very difficult to allocate reasonable labels because the threshold needs to be set manually; the neural network should be equipped with a high confidence level, which makes these methods susceptible to noise.

2.3. Self-supervised learning

Self-supervised is a new manner for unsupervised learning. It provides a surrogate supervised signal for feature learning (Gidaris et al., 2019). It uses the pretext tasks to automatically learn common representations and uses them as supervised signals to assist downstream tasks (Kolesnikov et al., 2019). In Computer Vision (Chen et al., 2020;

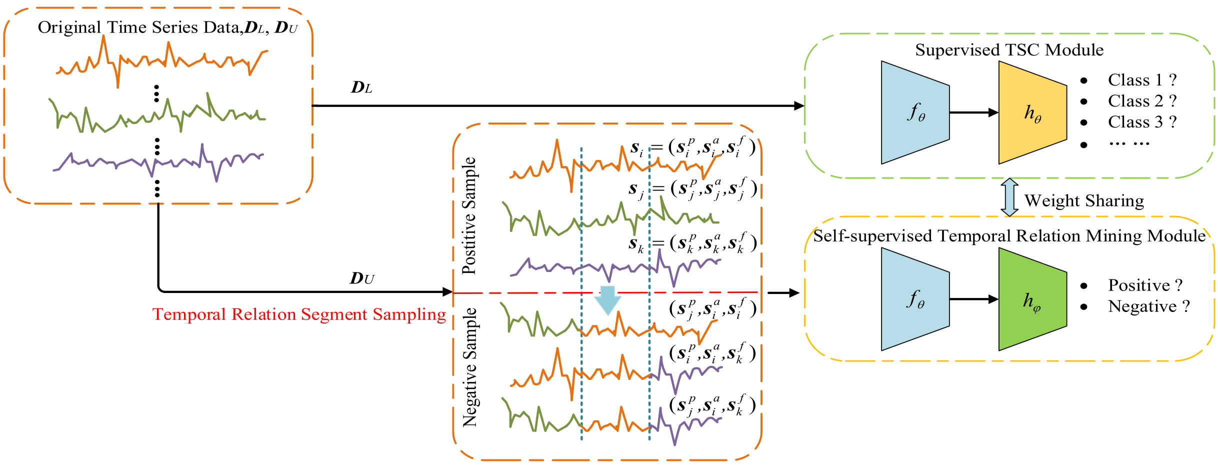

Fig. 3. Model framework.

Ren et al., 2020), self-supervised classification and detection methods outperform the supervised methods (Zhou and Iagnemma, 2010; Sharma et al., 2019). Moreover, Self-supervised learning has also been applied to TSC tasks of images, video, or audio. Ma et al. (2020) propose a Self-supervised Time-series Clustering Network based on Recurrent Neural Network (RNN) and spectral analysis. Banville et al. (2019) present a self-supervised strategy to learn the latent representations from multivariate time series in ElectroEncephaloGraph (EEG) by introducing two temporal contrastive learning tasks: "relative positioning" and "temporal shuffling". In the application field of remote sensing, Yuan and Lin (2020) use self-supervised pre-training to initialize a transformer-based network and improve the accuracy of Satellite Image TSC.

Recently, through self-supervised learning, some models improve the performance of semi-supervised learning (Zhai et al., 2019). One successful approach captures the ECG feature representations by predicting whether the time windows were sampled from the same temporal context or not (Banville et al., 2019). Time-series change point detection with self-supervised contrastive predictive coding (Deldari et al., 2021) and the self-supervised time series representation learning by inter-intra relational reasoning (Fan et al., 2020) also adopt similar strategies. Jawed et al. (2020) introduce a Multi-Task Learning (MTL) framework for exploiting the unlabeled training data by self-supervised learning and provide a strong supervised signal for feature learning. It enables the forecasting and classification tasks to share latent representations and learns high-order interactions automatically, but it ignores the inherent temporal relation. Our preliminary work, SemiTime (Fan et al., 2021), focuses on capturing the temporal relation features by sampling and combining time series segments, and achieves better TSC effects than the other methods. But SemiTime only considers the sampling of "past-future", which is not enough to obtain more refined temporal relations. On the basis of this, we further explore the more complex internal temporal relations among the time series data in this paper.

3. Method

SSTSC is mainly composed of a supervised TSC module and a self-supervised temporal relation mining module, as shown in Fig. 3. Self-supervised learning is used as an auxiliary task to learn the temporal relations of the time series, and jointly optimized with supervised learning by the shared weight backbone. The parameters are listed in Table 1.

The dataset is divided into two parts: the labeled data subset, $\boldsymbol{D}_L = \left\{ \boldsymbol{x} \,\middle|\, \boldsymbol{x} = \left(x_i^L, y_i^L\right) \right\}$, where y_i is the label, and the unlabeled data subset, $\mathbf{D}_U = \left\{ \mathbf{x} \,\middle|\, \mathbf{x} = \left(\mathbf{x}_i^U\right) \right\}$. $\boldsymbol{D}_L \subseteq \boldsymbol{D}_U$(Using labeled data as a part of unlabeled data can make the self-supervised module learn the temporal relation of the whole dataset well and help the supervised module perform the classification task more accurately. The labels are not considered in the self-supervised training). The input of the supervised TSC module is $\boldsymbol{D}_L$. In the self-supervised module, the input is $\boldsymbol{D}_U$. Because the unlabeled data contains labeled data, self-supervised learning also can learn the inherent temporal relations of the labeled time series, thereby helping supervised classification to achieve better performance.

Table 1
Parameter list.

Parameter	Description
$\boldsymbol{D}_L$	The labeled data subset
$\boldsymbol{D}_U$	The unlabeled data subset
$\mathbf{s}_i = [\mathbf{s}_i^p, \mathbf{s}_i^a, \mathbf{s}_i^f]$	A sample, s_i, and its past, anchor and future segments
$\mathbf{s}_i^+ = \left\{\mathbf{s}_i^p, \mathbf{s}_i^f\right\}$	The positive segments for an anchor, s_i
$\mathbf{s}_i^- = \left\{\mathbf{s}_j^p, \mathbf{s}_k^f\right\}, j, k \neq i$	The negative segments for an anchor, s_i
$\mathbf{z}_i^c$	The embeddings of labeled data
$\mathbf{z}_i^a$	The embeddings of anchor segments
$\mathbf{z}_i^p$	The embeddings of past segments
$\mathbf{z}_i^f$	The embeddings of future segments
$\mathbf{c}_i$	The classification results

3.1. Supervised TSC

The cores of the supervised TSC module, shown in Fig. 4, are the shared backbone encoder, f_θ, and the classifier, h_μ. f_θ is composed of a multi-layer 1D CNN with Batch Normalization (BN) and ReLU activation; h_μ is composed of a multi-layer FCN and Softmax. A pooling layer links f_θ and h_μ.

f_θ takes labeled time series data as input to extract the embeddings, $\mathbf{z}_i^c = f_\theta(\mathbf{x}_i), \mathbf{x} \in \mathbf{D}_L$. And then, with the assistance of the self-supervised temporal relation mining module, SSTSC gets the final classification results by the weight sharing mechanism: $\mathbf{c}_i = h_\mu(\mathbf{z}_i^c)$. The loss function is defined as the cross-entropy function:

$$L_c = -\frac{1}{|\mathbf{D}_L|} \sum_{i=1}^{|\mathbf{D}_L|} y_i^L \cdot \log\left(c_i\right) \tag{1}$$

3.2. Self-Supervised Temporal Relation Mining

For the unlabeled time series data, the Self-Supervised Temporal Relation Mining Module samples the temporal relation segments to

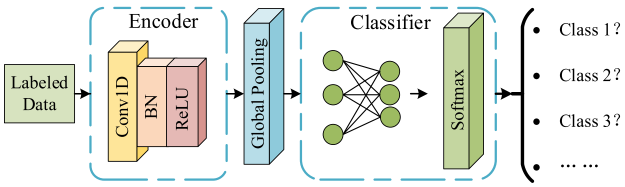

Fig. 4. Supervised TSC module.

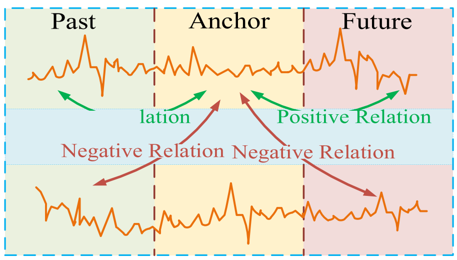

Fig. 5. Temporal relation construction.

perform the temporal relation prediction and semantic context learning and fully use the contextual representation of unlabeled time series data as the supervised signal to complete the final classification task.

Formally, each unlabeled data, $s_i \in \boldsymbol{D}_U$, is split into three parts:

$$\mathbf{s}_i = \left[\mathbf{s}_i^p, \mathbf{s}_i^a, \mathbf{s}_i^f\right]$$

where $\mathbf{s}_i^p$ denotes the past segment, $\mathbf{s}_i^a$ denotes the anchor segment, and $\mathbf{s}_i^f$ denotes the future segment.

Given an anchor segment $\mathbf{s}_i^a$, there are two types of positive segments, $\mathbf{s}_i^p$ and $\mathbf{s}_i^f$, which are collectively defined as $\mathbf{s}_i^+$; two types of negative segments, $\mathbf{s}_j^p$ and $\mathbf{s}_j^f$, $j \neq i$, which are collectively defined as $\mathbf{s}_i^-$. This is, the segments from the same sample are in positive temporal relation, $\widetilde{y} = 1$, while the segments from the different samples are in negative temporal relation, $\widetilde{y} = 0$, which can be seen in Fig. 5.

Based on this rule, SSTSC constructs four types of samples: one type of positive temporal relation samples (original samples) and three types of negative temporal relation samples, expressed as, $\left(\mathbf{s}_j^p, \mathbf{s}_i^a, \mathbf{s}_i^f\right)$, $\left(\mathbf{s}_i^p, \mathbf{s}_i^a, \mathbf{s}_j^f\right)$ and $\left(\mathbf{s}_j^p, \mathbf{s}_i^a, \mathbf{s}_k^f\right)$. The shared backbone encoder, f_θ, takes these anchor, positive and negative segments as inputs to get their embeddings, $\mathbf{z}_i^a = f_\theta\left(\mathbf{s}_i^a\right)$, $\mathbf{z}_i^p = f_\theta\left(\mathbf{s}_i^p\right)$, and $\mathbf{z}_i^f = f_\theta\left(\mathbf{s}_i^f\right)$. Then, the h_ϕ, composed of a multi-layer FCN and Sigmoid, predicts the temporal relations and learns the semantic context:

$$\mathbf{r}_i = \begin{cases} \mathbf{r}_i^+ = h_\phi([\mathbf{z}_i^p, \mathbf{z}_i^a, \mathbf{z}_i^f]), i = 1, 2, \ldots, |\mathbf{D}_U| \\ \mathbf{r}_i^- = h_\phi([\mathbf{z}_j^p, \mathbf{z}_i^a, \mathbf{z}_k^f]), j, k \neq i \end{cases} \tag{2}$$

where $\mathbf{r}_i^+$ represents the positive temporal relations, and the $\mathbf{r}_i^-$ represents the negative temporal relations. Also, a pooling layer links f_θ and h_ϕ. The loss function is defined as the binary cross-entropy function:

$$L_r = -\frac{1}{4\left|\mathbf{D}_U\right|} \sum_{i=1}^{4|\mathbf{D}_U|} \widetilde{y}_i \cdot \log\left(r_i\right) + \left(1 - \widetilde{y}_i\right) \cdot \left(1 - \log\left(r_i\right)\right) \tag{3}$$

The process of SSTSC is as follows. We use Adam Optimizer for model training, where η is the learning rate.

4. Experiments

4.1. Experimental setup

The experiments were written in Python 3.6, modeled in PyTorch 1.4 environment, and conducted on a Ubuntu-OS-based Server, whose main parameters were:

Table 2
Statistical information of the data sets.

Type	Dataset	Sample	Length	Class
Image	Herring	128	512	2
	Yoga	3300	426	2
Motion	CricketX	780	300	12
	CricketY	780	300	12
	CricketZ	780	300	12
ECG	ECG200	200	96	2
Sensor	XJTU	1320	1024	15
EEG	EpilepticSeizure	11500	178	5

CPU: Intel(R) Xeon(R) CPU E5-2640(16-Core, 2.4 GHz)
GPU: 8× NVIDIA GeForce RTX 2080Ti
Memory: 128G RAM

The experiments selected different time series datasets with different characteristics, including six datasets from UCRARCHIVE_2018: CricketX, CricketY, CricketZ, ECG200, Herring and Yoga. In addition, an actual bearing dataset, XJTU, and an EEG dataset, EpilepticSeizure, were selected. The basic statistical information of these datasets is shown in Table 2. For each dataset, we set train-validation-test split as 60%-20%-20%.

Several currently known TSC models are selected for experimental comparison with SSTSC.

(1) Supervised: It uses the same supervised framework of SSTSC, and is trained only on labeled data.

(2) Pseudo-Label (Lee, 2013): It is a semi-supervised learning method that uses the pseudo-label generated from unlabeled data to enlarge the training set for supervised training. We use it for TSC.

(3) Π-Model (Laine and Aila, 2016): It is a semi-supervised learning method that uses self-ensembling to form a consensus prediction of the unknown labels under different regularization and input augmentation conditions. We use it for TSC.

(4) MTL (Jawed et al., 2020): It is a multi-task semi-supervised TSC framework that uses self-supervised learning to forecast time series values, so as to assist supervised learning to complete TSC tasks. However, Its self-supervised learning does not consider temporal relations, which is the most significant difference from SSTSC, because SSTSC's self-supervised learning is a relational reasoning task, which is to learn the inherent temporal relations of the time series.

(5) SemiTime (Fan et al., 2021): It is a semi-supervised TSC method that divides the time series data into two segments, for predicting the temporal relations based on self-supervised and serves as an auxiliary task for semi-supervised TSC. SemiTime is the preliminary work of the SSTSC in this paper.

In order to ensure fairness, we use the same 4-layer 1D CNN for time-series feature extraction of these methods. The core ideas and other modules of these baselines remain as they are. In SSTSC, the learning rate is 0.01. The number of neurons in each CNN layer is 8, 16, 32 and 64, respectively. The layer number of FCN is 2, the number of neurons in each FCN layer is 256. The batch-size is 128. The other parameters settings of baselines were based on the settings in their respective References, and fine-tuned with the optimal test results. We train all models with 1000 iterations with an early-stopping callback of 200 patience iterations to monitor the validation metric and stop the training when no improvement is observed. All the experimental results of the mean and standard deviation by the experiment running over five seeds. During training, we use data augmentations (magnitude warping and time warping (Um et al., 2017)) for all methods.

This paper adopts the "accuracy (ACC)" as the evaluation metric, which is typically used in the related research field.

Algorithm 1 SSTSC mini-batch training process

INPUT: $\boldsymbol{D}_L$ and $\boldsymbol{D}_U$

OUTPUT: TSC results

BEGIN:

FOR each epoch DO

  FOR each labeled minibatch, $\boldsymbol{B}_L \subset \boldsymbol{D}_L$, DO //Supervised TSC module

    $z_i^c = f_\theta(\boldsymbol{x}_{i\in|\boldsymbol{B}_L|})$ //Embeddings of labeled inputs

    $\boldsymbol{c}_i = h_\mu(\boldsymbol{z}^c_{i\in|\boldsymbol{B}_L|})$ //Label classification

    $L_c = -\frac{1}{|\boldsymbol{D}_L|}\sum_{i=1}^{|\boldsymbol{D}_L|} y_i^L \cdot \log(\boldsymbol{c}_i)$

    $\theta = \theta - \eta \triangledown_\theta L_c, \mu = \mu \triangledown_\mu L_c$ //Update model

  end FOR

  FOR each unlabeled minibatch, $\boldsymbol{B}_U \subset \boldsymbol{D}_U$, DO //Self-supervised temporal relation mining module

    $\boldsymbol{z}_i^a = f_\theta(\boldsymbol{s}^a_{i\in|\boldsymbol{B}_U|})$ //Embeddings of anchor segments

    $\boldsymbol{z}_i^p = f_\theta(\boldsymbol{s}^p_{i\in|\boldsymbol{B}_U|})$ //Embeddings of positive segments

    $\boldsymbol{z}_i^f = f_\theta(\boldsymbol{s}^f_{i\in|\boldsymbol{B}_U|})$ //Embeddings of negative segments

    $\boldsymbol{r}_i^+ = h_\varphi([\boldsymbol{z}_i^p, \boldsymbol{z}_i^a, \boldsymbol{z}_i^f])$

    $\boldsymbol{r}_i^- = h_\varphi([\boldsymbol{z}_j^p, \boldsymbol{z}_i^a, \boldsymbol{z}_k^f]), j, k \neq i$

    $L_r = -\frac{1}{4|\boldsymbol{D}_U|}\sum_{i=1}^{4|\boldsymbol{D}_U|} \tilde{y}_i \cdot \log(\boldsymbol{r}_i) + (1-\tilde{y}_i)\cdot(1-\log(\boldsymbol{r}_i))$

    $\theta = \theta - \eta \triangledown_\theta L_r, \mu = \mu \triangledown_\varphi L_r$ //Update model

    end FOR

  end FOR

RETURN f_θ and h_μ

END.

4.2. Experimental results and analysis

We perform five different experiments on eight different datasets.

(1) Parameter sensitivity experiment: The purpose is mainly to investigate the effects of the temporal relation mining module under different split ratios of "past-anchor-future" segments, "$\alpha - \beta - \alpha$" on TSC, where $\alpha \in (0,0.5]$, β=1-2α.

(2) Ablation experiment: The purpose is mainly to test the actual function of different components in SSTSC, to highlight the effectiveness of the combination of the supervised module and the self-supervised module.

(3) Comparative experiment: The purpose is mainly to demonstrate that SSTSC has better overall TSC performance than the other baselines.

(4) Classification efficiency experiment: The purpose is mainly to highlight that SSTSC can significantly improve the TSC performance on the basis of better or comparable efficiency to the other baselines.

(5) Visualization experiment: The purpose is mainly to show that SSTSC can get better classification boundaries which demonstrates the reason SSTSC can perform better classification performance than the other baselines.

(1) Parameter sensitivity experiment

We use different split ratios on different datasets (the label ratio in the training set is 10%) to verify the temporal relation prediction ACC of Self-Supervised Temporal Relation Mining and the classification ACC of SSTSC. As shown in Fig. 6, the temporal relation prediction ACCs are relatively better when $\alpha = \beta = 0.3$, $\gamma = 0.4$. Therefore, we set $\alpha = \beta$ to carry out the TSC experiment and the results are shown in Fig. 7. We find that the classification results are relatively better when α is 0.2 or 0.3. Therefore, "$\alpha - \gamma - \beta$" is set as "0.3–0.4-0.3" in the subsequent experiments.

(2) Ablation experiment

We continue to use the environment and process of the first experiment to test the classification results of the model with/without Self-supervised Temporal Relation Mining (i.e., SSTSC and Supervised), and the results are shown in Fig. 8. The blue bar represents the prediction ACC of temporal relation mining; The red line represents the classification ACC of the SSTSC, while the yellow dotted line represents the classification ACC of the Supervised module. We can find that SSTSC performs better than the Supervised baseline on all datasets except ECG200 and Yoga. But, the worse results of SSTSC than that of Supervised only appear in the case of unreasonable split ratios on the ECG200 and Yoga, which can explain that supervised learning is focused on specific mapping, while SSTSC can accurately mine the latent temporal relations of time series data with reasonable split ratios, so that the classification task can learn the inherent laws of time series and obtain better performance. These results also show that the temporal relation mining plays a significant role in improving the classification performance of SSTSC.

(3) Comparative experiment

In this experiment, we randomly select partial samples (10%, 20%, 40%, and 100%) from the training set as labeled data, and regard the whole training set as unlabeled data. The results are shown in Table 2. SSTSC outperforms the baselines on all datasets, which can verify that the temporal relation of "past-anchor-future" is a universal implicit relationship in time series data. The baselines are not ideal on some datasets due to their shortcomings.

(i) The classification results of the SSTSC and the Supervised method are consistent with the results of the **Ablation experiment**.

(ii) Compared with the pseudo-label and Π-Model methods, SSTSC improves the learning ability by predicting the temporal relations

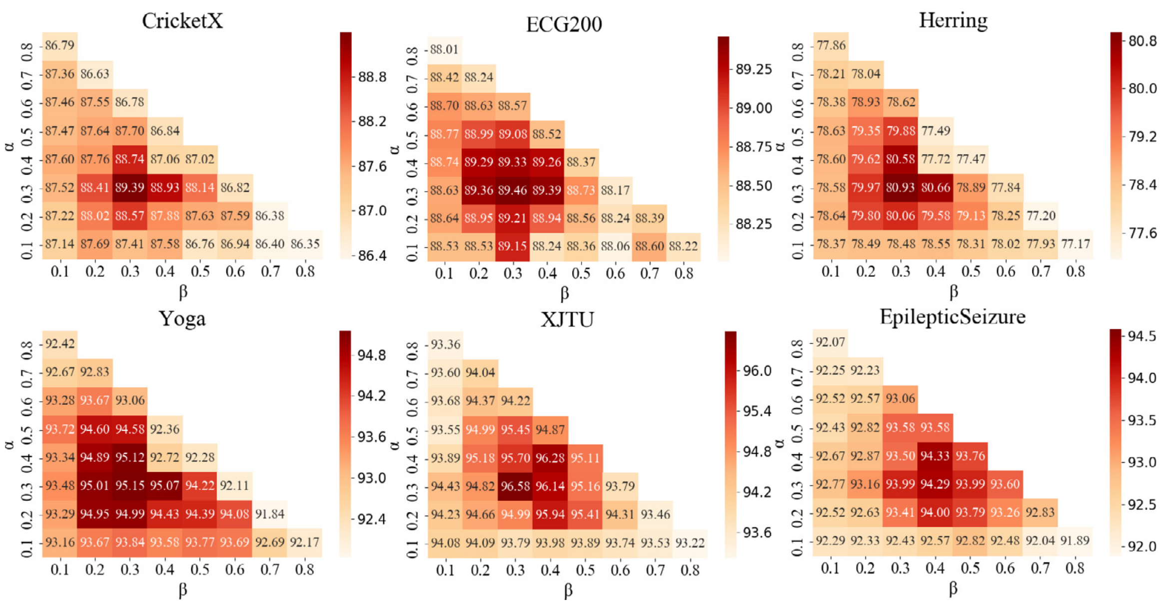

Fig. 6. Temporal relation prediction results of Self-Supervised Temporal Relation Mining with different "$\alpha-\gamma-\beta$"'s on different datasets (label ratio = 10%).

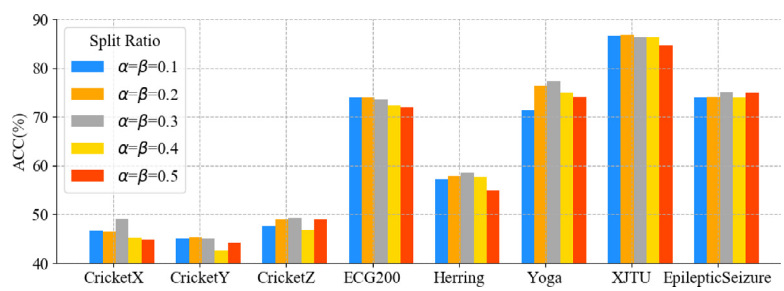

Fig. 7. Classification results of SSTSC with different "past-anchored-future" split ratios on different datasets (label ratio = 10%).

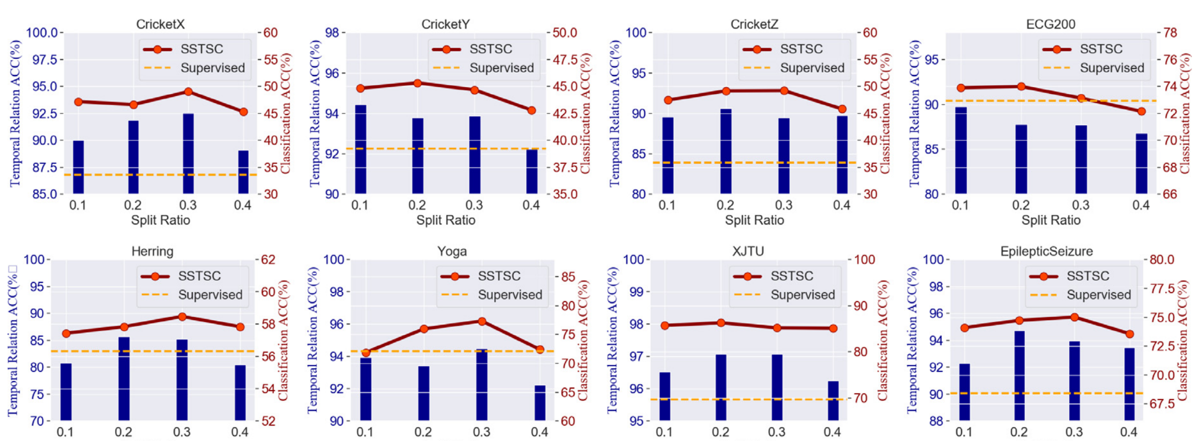

Fig. 8. Ablation experimental results of SSTSC with different "past-anchor-future" split ratios on different datasets (label ratio = 10%).

among time series, and avoids the problems of manually setting the thresholds and being easily affected by noise.

(iii) When the training set is 100% labeled data, MTL is inferior to the Supervised method except on CricketX, XJTU, and EpilepticSeizure, and SemiTime is inferior to the Supervised method on CricketY and Yoga. It highlights that MTL does not consider temporal relation, so it is inferior to the Supervised method on some datasets; SemiTime considers temporal relation and gets better effects than MTL, but it is still inferior to the Supervised method on the CricketY and Yoga datasets because of the insufficient mining by "past-future" split manner. After fully considering temporal relation mining, SSTSC achieves the ideal effects.

(iv) Throughout all split ratios, SemiTime is superior to MTL on most datasets, indicating that self-supervised learning that considers the inherent temporal relation are more conducive to semi-supervised TSC task. However, on ECG200 and Herring, the performance of MTL is similar to that of SemiTime. One possible reason is that the number of samples is too small, and MTL and SemiTime cannot fully learn useful contextual representations. In contrast, SSTSC consistently outperforms the MTL and SemiTime even on the small datasets, which

Table 3
Classification results on different data sets.

Label Ratio	**10%**	**20%**	**40%**	**100%**	**10%**	**20%**	**40%**	**100%**
Dataset	**CricketX**				**CricketY**			
Supervised	33.62±0.95	38.79±2.08	52.64±2.53	62.98±2.01	39.21±2.21	46.80±2.15	58.70±1.58	71.25±1.93
Pseudo-Label	38.87±2.26	44.44±2.91	53.39±2.18	-	41.24±2.33	50.69±1.98	57.92±3.20	-
Π-Model	38.61±2.29	48.18±2.07	54.73±1.04	-	42.67±1.33	54.29±1.53	56.17±1.84	-
MTL	40.94±1.97	50.12±1.22	55.10±1.12	63.58±1.72	38.33±1.55	47.56±1.35	51.55±2.22	63.67±2.54
SemiTime	44.88±3.13	51.61±0.66	58.71±2.78	65.66±1.58	44.26±2.47	53.74±4.12	60.80±2.84	70.15±3.48
SSTSC	**49.03±1.29**	**56.10±2.49**	**60.68±2.57**	**68.85±1.90**	**45.33±0.97**	**54.35±0.97**	**61.81±2.20**	**71.73±2.76**
Range	*15.41*	*17.31*	*8.04*	*5.87*	*7.00*	*7.55*	*10.26*	*8.06*
Dataset	**CricketZ**				**ECG200**			
Supervised	35.84±1.96	44.09±1.89	54.06±1.93	65.21±2.24	72.92±4.48	75.18±2.76	78.10±2.34	81.49±2.71
Pseudo-Label	43.11±1.71	51.81±0.49	58.91±1.12	-	73.33±3.50	74.15±3.07	77.64±2.41	-
Π-Model	46.82±1.33	55.82±1.82	62.86±1.88	-	73.13±2.44	75.33±2.18	77.33±1.93	-
MTL	39.73±1.35	45.38±1.21	49.33±1.57	61.93±1.93	72.10±2.92	75.33±3.01	77.33±3.01	80.10±2.02
SemiTime	48.98±1.60	56.41±0.94	62.46±1.45	69.44±1.96	72.00±2.70	75.90±3.42	77.02±2.54	82.26±1.96
SSTSC	**49.23±1.39**	**57.49±1.59**	**64.44±2.22**	**70.47±1.54**	**74.00±4.40**	**76.77±3.51**	**78.72±0.78**	**82.31±2.02**
Range	*13.39*	*13.40*	*15.11*	*8.54*	*2.00*	*2.62*	*1.70*	*2.21*
Dataset	**Herring**				**Yoga**			
Supervised	56.32±1.48	56.08±1.72	56.80±1.70	57.52±4.69	72.14±1.65	80.03±1.33	81.15±0.71	91.34±0.66
Pseudo-Label	53.12±2.69	55.76±1.94	56.88±2.87	-	70.56±0.99	78.36±1.02	85.26±1.02	-
Π-Model	56.08±3.36	56.24±1.55	57.76±2.80	-	65.82±1.11	67.56±0.74	67.06±1.68	-
MTL	54.64±2.41	56.00±2.18	56.32±2.32	55.84±3.72	69.08±0.73	73.78±1.31	79.75±0.62	87.44±1.12
SemiTime	54.88±3.14	56.12±3.14	56.32±3.38	58.40±5.52	74.05±1.80	78.50±0.70	84.58±0.55	90.28±1.09
SSTSC	**58.48±1.42**	**58.56±2.33**	**59.52±1.81**	**60.16±3.44**	**77.34±0.79**	**82.25±1.29**	**86.86±1.12**	**91.72±0.58**
Range	*5.36*	*2.80*	*3.20*	*4.32*	*11.52*	*14.69*	*19.80*	*4.28*
Dataset	**XJTU**				**EpilepticSeizure**			
Supervised	69.71±1.96	83.32±1.59	94.03±1.56	97.92±0.61	68.40±0.43	70.77±0.70	73.49±0.60	77.77±1.13
Pseudo-Label	74.88±2.78	85.19±1.82	93.97±2.79	-	68.57±0.50	72.92±0.48	74.60±0.65	-
Π-Model	75.96±0.52	85.93±0.91	95.03±1.34	-	69.60±0.34	71.58±0.64	74.54±0.55	-
MTL	73.22±1.86	86.64±1.78	94.02±1.65	98.15±1.04	68.71±0.94	73.17±0.81	74.77±0.75	78.53±0.62
SemiTime	84.61±1.39	93.93±0.49	97.79±0.33	98.46±0.25	74.86±0.42	75.54±0.63	77.01±0.79	79.26±1.20
SSTSC	**86.29±0.82**	**94.79±0.81**	**97.96±0.21**	**99.24±0.28**	**75.02±0.46**	**75.84±0.45**	**77.42±0.79**	**79.94±0.50**
Range	*16.58*	*11.47*	*3.99*	*1.32*	*6.62*	*5.07*	*3.93*	*2.17*

Table 4
Friedman test results (significance level of 0.05, label ratio=10%).

	Rank
Supervised	4.875
Pseudo-Label	4.375
Π-Model	4.125
MTL	3.750
SemiTime	2.750
SSTSC	**1.125**

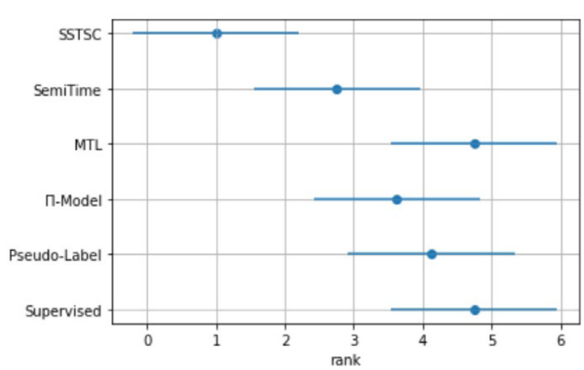

Fig. 9. Friedman test results (significance level of 0.05, label ratio = 10%).

means that through the construction of more types and quantities of negative temporal relation samples by the three-segment split manner of "past-anchor-future", SSTSC can predict higher-quality temporal relations than SemiTime, which uses the two-segment split manner of "past-future".

(v) Herring, Yoga and ECG200 are binary datasets; CricketX, CricketY, CricketZ, XJTU and EpilepticSeizure are multi-class datasets (shown in Table 1). SSTSC gets the best results on both binary datasets and multi-class datasets. Moreover, compared with the results of binary datasets, SSTSC has more significant advantages over baselines on multi-class datasets as a whole. It indicates that SSTSC can mine the differences between classes more precisely through temporal relation analysis of "past-anchor-future".

(vi) On the basis of the classification results, combined with the results of Friedman tests (shown in Table 4 and Fig. 9), we can also find that SSTSC can achieve ideal results on datasets with different characteristics by accurately extracting the temporal relation features.

(4) Classification efficiency experiment

In this experiment, we randomly select 10% samples from the training set as the labeled data and use the whole training set as the unlabeled data for model training. The classification time (unit: s) of each batch-size is recorded, and summarized in Fig. 10. Generally, the efficiency of STSSC on each dataset is in the middle or upper reaches. Compared with the classification results in Table 3, SSTSC has achieved the best classification performance while considering efficiency.

(5) Visualization experiment

We continue to use the environment and process of the "**Comparative experiment**" and make a visualization comparison of latent representations obtained by the different models on the XJTU and EpilepticSeizure. The latent representations extracted from each model are as the input of the t-SNE tool (Van der Maaten and Hinton, 2008) to generate 2D visualization results shown in Fig. 11. The dots in each color represent a class of samples. We can see that SSTSC can obtain better classification boundaries than other baselines. Moreover, the visualization results are consistent with the classification results in Table 3.

These results indicate that the clustering ability of the TSC task can be improved by obtaining higher-quality semantic context and using it as a supervised signal to assist supervised learning and get better classification boundaries. It also confirms that the predicting "past-anchor-future" temporal relations plays a decisive role in improving the TSC performance of SSTSC.

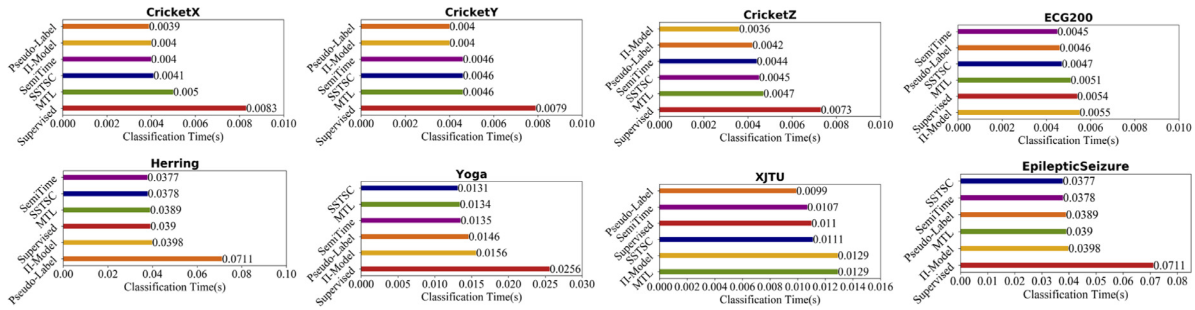

Fig. 10. Classification efficiency results on different datasets (label ratio = 10%).

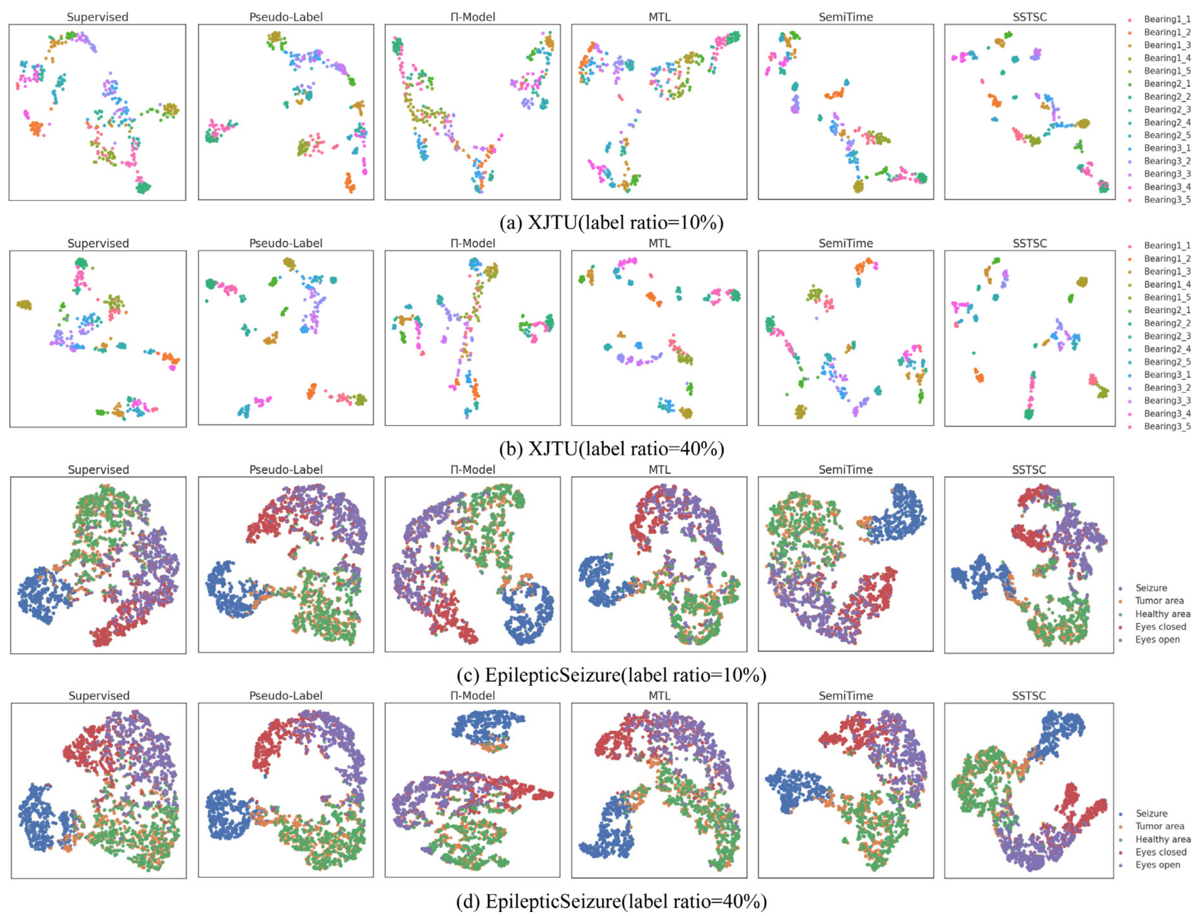

Fig. 11. Visualization results on the XJTU and EpilepticSeizure datasets.

5. Conclusion

We propose the SSTSC to predict temporal relations accurately and self-supervised-learn higher-quality semantic context from unlabeled time series data by sampling and constructing the positive and negative temporal relations of "past-anchor-future". Using the semantic context as the self-generated supervised signal can better assist semi-supervised TSC by training with labeled and unlabeled time series jointly. Comprehensive experimental results show that SSTSC performs better than the baselines on multiple real-world datasets. We will continue to design a more effective and accurate temporal relation sampling strategy to further improve the performance and apply it to multivariate time series classification.

CRediT authorship contribution statement

Liang Xi: Conceptualization, Methodology, Validation, Formal analysis, Investigation, Resources, Supervision, Visualization, Project administration, Funding acquisition, Writing – review & editing. **Zichao Yun:** Conceptualization, Methodology, Investigation, Software, Data curation, Writing – original draft. **Han Liu:** Software, Writing – review & editing. **Ruidong Wang:** Software, Writing – review & editing. **Xunhua Huang:** Software, Writing – review & editing. **Haoyi Fan:** Methodology, Software.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

We have shared the source code and datasets on our personal website: https://github.com/mrxiliang/sstsc.git.

Acknowledgments

This work was supported by the Natural Science Foundation of China under Grant 61172168, and Natural Science Foundation of Heilongjiang Province, China under Grant F2018019.

References

Bagnall, A., Lines, J., Bostrom, A., Large, J., Keogh, E., 2017. The great time series classification bake off: a review and experimental evaluation of recent algorithmic advances. Data Min. Knowl. Discov. 31 (3), 606–660. http://dx.doi.org/10.1007/s10618-016-0483-9.

Banville, H., Albuquerque, I., Hyvärinen, A., Moffat, G., Engemann, D.A., Gramfort, A., 2019. Self-supervised representation learning from electroencephalography signals. In: The 29th International Workshop on Machine Learning for Signal Processing, Pittsburgh, USA. http://dx.doi.org/10.1109/MLSP.2019.8918693, 8918693.

Berthelot, D., Carlini, N., Goodfellow, I., Papernot, N., Oliver, A., Raffel, C.A., 2019. Mixmatch: A holistic approach to semi-supervised learning. In: The 33rd Conference on Neural Information Processing Systems, Vancouver, Canada. pp. 1–14, arXiv:1905.02249.

Chen, L., Chen, D., Yang, F., Sun, J., 2021. A deep multi-task representation learning method for time series classification and retrieval. Inform. Sci. 555, 17–32. http://dx.doi.org/10.1016/j.ins.2020.12.062.

Chen, T., Kornblith, S., Norouzi, M., Hinton, G., 2020. A simple framework for contrastive learning of visual representations. In: The 37th International Conference on Machine Learning. pp. 1597–1607, arXiv:2002.05709.

Deldari, S., Smith, D.V., Xue, H., Salim, F.D., 2021. Time series change point detection with self-supervised contrastive predictive coding. In: 2021 World Wide Web Conference, New York, USA. pp. 3124–3135. http://dx.doi.org/10.1145/3442381.3449903.

Esling, P., Agon, C., 2012. Time-series data mining. ACM Comput. Surv. 45 (1), 1–34. http://dx.doi.org/10.1145/2379776.2379788.

Fan, H., Zhang, F., Gao, Y., 2020. Self-supervised time series representation learning by inter-intra relational reasoning. arXiv preprint, arXiv:2011.13548.

Fan, H., Zhang, F., Wang, R., Huang, X., Li, Z., 2021. Semi-supervised time series classification by temporal relation prediction. In: 2021-2021 IEEE International Conference on Acoustics, Speech and Signal Processing, Toronto, Canada. pp. 3545–3549. http://dx.doi.org/10.1109/ICASSP39728.2021.9413883.

Fawaz, H.I., Forestier, G., Weber, J., Idoumghar, L., Muller, P.A., 2018. Transfer learning for time series classification. In: The 4th IEEE International Conference on Big Data, Seattle, USA. pp. 1367–1376. http://dx.doi.org/10.1109/BigData.2018.8621990.

Gidaris, S., Bursuc, A., Komodakis, N., Pérez, P., Cord, M., 2019. Boosting few-shot visual learning with self-supervision. In: The 2019 IEEE/CVF International Conference on Computer Vision, Seoul, Korea. pp. 8059–8068. http://dx.doi.org/10.1109/ICCV.2019.00815.

Hailat, Z., Komarichev, A., Chen, X.W., 2018. Deep semi-supervised learning. In: The 24th International Conference on Pattern Recognition, Beijing, China. pp. 2154–2159. http://dx.doi.org/10.1109/ICPR.2018.8546327.

He, K., Fan, H., Wu, Y., Xie, S., Girshick, R., 2020. Momentum contrast for unsupervised visual representation learning. In: The 33th IEEE/CVF Conference on Computer Vision and Pattern Recognition, Seattle, USA. pp. 9729–9738. http://dx.doi.org/10.1109/CVPR42600.2020.00975.

Husain, M., Meena, S.M., 2019. Multimodal fusion of speech and text using semi-supervised LDA for indexing lecture videos. In: The 25th National Conference on Communications, Bangalore, India. http://dx.doi.org/10.1109/NCC.2019.8732253, 8732253.

Ismail Fawaz, H., Forestier, G., Weber, J., Idoumghar, L., Muller, P.A., 2019. Deep learning for time series classification: a review. Data Min. Knowl. Discov. 33 (4), 917–963. http://dx.doi.org/10.1007/s10618-019-00619-1.

Jawed, S., Grabocka, J., Schmidt-Thieme, L., 2020. Self-supervised learning for semi-supervised time series classification. In: 24th Pacific-Asia Conference on Knowledge Discovery and Data Mining, Singapore. pp. 499–511. http://dx.doi.org/10.1007/978-3-030-47426-3_39.

Jing, L., Tian, Y., 2020. Self-supervised visual feature learning with deep neural networks: A survey. IEEE Trans. Pattern Anal. Mach. Intell. 43 (11), 4037–4058. http://dx.doi.org/10.1109/TPAMI.2020.2992393.

Karim, F., Majumdar, S., Darabi, H., Chen, S., 2017. LSTM fully convolutional networks for time series classification. IEEE Access 6, 1662–1669. http://dx.doi.org/10.1109/ACCESS.2017.2779939.

Keogh, E.J., Pazzani, M.J., 2001. Derivative dynamic time warping. In: The 1st SIAM International Conference on Data Mining, Chicago, USA. pp. 1–11. http://dx.doi.org/10.1137/1.9781611972719.1.

Kolesnikov, A., Zhai, X., Beyer, L., 2019. Revisiting self-supervised visual representation learning. In: The 37th IEEE/CVF Conference on Computer Vision and Pattern Recognition, Long Beach, USA. pp. 1920–1929. http://dx.doi.org/10.1109/CVPR.2019.00202.

Kumar, N., Awate, S.P., 2020. Semi-supervised robust mixture models in RKHS for abnormality detection in medical images. IEEE Trans. Image Process. 29, 4772–4787. http://dx.doi.org/10.1109/TIP.2020.2975958.

Laine, S., Aila, T., 2016. Temporal ensembling for semi-supervised learning. In: The 5th International Conference on Learning Representations. Toulon, France, arXiv:1610.02242.

Large, J., Bagnall, A., Malinowski, S., Tavenard, R., 2019. On time series classification with dictionary-based classifiers. Intell. Data Anal. 23 (5), 1073–1089. http://dx.doi.org/10.3233/IDA-184333.

Lee, D.H., 2013. Pseudo-label: The simple and efficient semi-supervised learning method for deep neural networks. In: 2013 Workshop on Challenges in Representation Learning. Atlanta, USA, pp. 1–6.

Li, Y., Song, H., Miao, B., Zhang, W., Xiong, Q., 2021. Improved multiscale weighted-dispersion entropy and its application in fault diagnosis of train bearing. Meas. Sci. Technol. 32 (7), 075002. http://dx.doi.org/10.1088/1361-6501/abdc98.

Lines, J., Bagnall, A., 2015. Time series classification with ensembles of elastic distance measures. Data Min. Knowl. Discov. 29 (3), 565–592. http://dx.doi.org/10.1007/s10618-014-0361-2.

Ma, Q., Li, S., Zhuang, W., Wang, J., Zeng, D., 2020. Self-supervised time series clustering with model-based dynamics. IEEE Trans. Neural Netw. Learn. Syst. 32 (9), 3942–3955. http://dx.doi.org/10.1109/TNNLS.2020.3016291.

Moon, J., Hossain, M.B., Chon, K.H., 2021. AR and ARMA model order selection for time-series modeling with ImageNet classification. Signal Process. 183, 108026. http://dx.doi.org/10.1016/j.sigpro.2021.108026.

Nweke, H.F., Teh, Y.W., Al-Garadi, M.A., Alo, U.R., 2018. Deep learning algorithms for human activity recognition using mobile and wearable sensor networks: State of the art and research challenges. Expert Syst. Appl. 105, 233–261. http://dx.doi.org/10.1016/j.eswa.2018.03.056.

Pavllo, D., Feichtenhofer, C., Grangier, D., Auli, M., 2019. 3D human pose estimation in video with temporal convolutions and semi-supervised training. In: The 37th IEEE/CVF Conference on Computer Vision and Pattern Recognition, Long Beach, USA. pp. 7753–7762. http://dx.doi.org/10.1109/CVPR.2019.00794.

Rasmus, A., Berglund, M., Honkala, M., Valpola, H., Raiko, T., 2015. Semi-supervised learning with ladder networks. In: The 29th Conference on Neural Information Processing Systems, Montreal, Canada. pp. 3546–3554, arXiv:1507.02672.

Ren, X., Li, H., Huang, Z., Chen, Q., 2020. Self-supervised dance video synthesis conditioned on music. In: The 28th ACM International Conference on Multimedia, New York, USA. pp. 46–54. http://dx.doi.org/10.1145/3394171.3413932.

Ribeiro, A.H., Ribeiro, M.H., Paixão, G.M., Oliveira, D.M., Gomes, P.R., Canazart, J.A., et al., 2020. Automatic diagnosis of the 12-lead ECG using a deep neural network. Nature Commun. 11 (1), 1–9. http://dx.doi.org/10.1038/s41467-020-15432-4.

Rizve, M.N., Duarte, K., Rawat, Y.S., Shah, M., 2021. In defense of pseudo-labeling: An uncertainty-aware pseudo-label selection framework for semi-supervised learning. arXiv preprint, arXiv:2101.06329.

Rodríguez, J.J., Alonso, C.J., Maestro, J.A., 2005. Support vector machines of interval-based features for time series classification. Knowl.-Based Syst. 18 (4–5), 171–178. http://dx.doi.org/10.1016/j.knosys.2004.10.007.

Sharma, V., Tapaswi, M., Sarfraz, M.S., Stiefelhagen, R., 2019. Self-supervised learning of face representations for video face clustering. In: The 14th IEEE International Conference on Automatic Face & Gesture Recognition, Lille, France. http://dx.doi.org/10.1109/FG.2019.8756609, 8756609.

Um, T.T., Pfister, F.M., Pichler, D., Endo, S., Lang, M., Hirche, S., et al., 2017. Data augmentation of wearable sensor data for parkinson's disease monitoring using convolutional neural networks. In: The 19th ACM International Conference on Multimodal Interaction, Glasgow, United Kingdom. pp. 216–220. http://dx.doi.org/10.1145/3136755.3136817.

Van der Maaten, L., Hinton, G., 2008. Visualizing data using t-SNE. J. Mach. Learn. Res. 9 (11), 2579–2605.

Verbesselt, J., Hyndman, R., Newnham, G., Culvenor, D., 2010. Detecting trend and seasonal changes in satellite image time series. Remote Sens. Environ. 114 (1), 106–115. http://dx.doi.org/10.1016/j.rse.2009.08.014.

Wang, Z., Yan, W., Oates, T., 2017. Time series classification from scratch with deep neural networks: A strong baseline. In: The 14th International Joint Conference on Neural Networks, Anchorage, USA. pp. 1578–1585. http://dx.doi.org/10.1109/IJCNN.2017.7966039.

Xing, Z., Pei, J., Yu, P.S., 2012. Early classification on time series. Knowl. Inf. Syst. 31 (1), 105–127. http://dx.doi.org/10.1007/s10115-011-0400-x.

Ye, L., Keogh, E., 2011. Time series shapelets: a novel technique that allows accurate, interpretable and fast classification. Data Min. Knowl. Discov. 22 (1), 149–182. http://dx.doi.org/10.1007/s10618-010-0179-5.

Yuan, Y., Lin, L., 2020. Self-supervised pretraining of transformers for satellite image time series classification. IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens. 14, 474–487. http://dx.doi.org/10.1109/JSTARS.2020.3036602.

Zhai, X., Oliver, A., Kolesnikov, A., Beyer, L., 2019. S4l: Self-supervised semi-supervised learning. In: The 17th IEEE/CVF International Conference on Computer Vision, Seoul, Korea. pp. 1476–1485. http://dx.doi.org/10.1109/ICCV.2019.00156.

Zhang, C., Cheng, J., Tian, Q., 2019. Unsupervised and semi-supervised image classification with weak semantic consistency. IEEE Trans. Multimed. 21 (10), 2482–2491. http://dx.doi.org/10.1109/TMM.2019.2903628.

Zhao, B., Lu, H., Chen, S., Liu, J., Wu, D., 2017. Convolutional neural networks for time series classification. J. Syst. Eng. Electron. 28 (1), 162–169. http://dx.doi.org/10.21629/JSEE.2017.01.18.

Zhou, S., Iagnemma, K., 2010. Self-supervised learning method for unstructured road detection using fuzzy support vector machines. In: The 23rd IEEE/RSJ International Conference on Intelligent Robots and Systems, Taipei, Taiwan. pp. 1183–1189. http://dx.doi.org/10.1109/IROS.2010.5650300.