计算机科学
COMPUTER SCIENCE

基于对比学习的时间序列聚类方法

杨博, 罗嘉琛, 宋艳涛, 吴宏涛, 彭甫镕

引用本文

杨博, 罗嘉琛, 宋艳涛, 吴宏涛, 彭甫镕. 基于对比学习的时间序列聚类方法[J]. 计算机科学, 2024, 51(2): 63-72.

YANG Bo, LUO Jiachen, SONG Yantao, WU Hongtao, PENG Furong. Time Series Clustering Method Based on Contrastive Learning [J]. Computer Science, 2024, 51(2): 63-72.

计算机科学 COMPUTER SCIENCE
http://www.jsjkx.com
DOI:10.11896/jsjkx.221200038

基于对比学习的时间序列聚类方法

杨　博[1,2]　罗嘉琛[1,2]　宋艳涛[1,2]　吴宏涛[3]　彭甫镕[1,2]
1 山西大学大数据科学与产业研究院　太原 030006
2 山西大学计算机与信息技术学院　太原 030006
3 山西省交通科技研发有限公司　太原 030006
(yangbo981205@gmail.com)

摘　要　现有深度聚类方法严重依赖于复杂的特征提取网络和聚类算法,难以直观地定义时间序列的相似性。使用对比学习的方法可以从正负样本数据的角度定义时间序列的区间相似性,并对特征提取和聚类进行联合优化。基于对比学习的思想,提出了一种不依赖于复杂表示网络的时间序列聚类模型。同时,为解决现有时间序列数据增强方法难以描述时间序列的变换不变性的问题,提出了一种基于时间序列形状特征的数据增强方法,在忽略数据时域特征情况下捕捉序列的相似性。模型通过设置不同的形状转换参数构造正负样本对,学习特征表示并投影到特征空间,在实例级对比和聚类级对比层面利用交叉熵损失最大化正样本对相似性,最小化负样本对相似性,实现了端到端的联合学习表示和聚类分配。在32个UCR中的数据集上进行了大量实验,结果表明该模型可以在不依赖于特定表示学习网络的情况下得到与现有方法相当或优于现有方法的聚类结果。

关键词:时间序列聚类;对比学习;数据增强;表示学习;联合优化
中图分类号　TP183

Time Series Clustering Method Based on Contrastive Learning

YANG Bo[1,2],LUO Jiachen[1,2],SONG Yantao[1,2],WU Hongtao[3] and PENG Furong[1,2]
1 Institute of Big Data Science and Industry,Shanxi University,Taiyuan 030006,China
2 School of Computer and Information Technology,Shanxi University,Taiyuan 030006,China
3 Shanxi Transportation Technology R&D Co.,Ltd,Taiyuan 030006,China

Abstract　It is difficult to intuitively define the similarity between time series by deep clustering methods which rely heavily on complex feature extraction networks and clustering algorithms. Contrastive learning can define the interval similarity of time series from the perspective of positive and negative sample data and jointly optimize feature extraction and clustering. Based on the contrastive learning,this paper proposes a time series clustering model that does not rely on complex representation networks. In order to solve the problem that the existing time series data enhancement methods cannot describe the transformation invariance of time series,this paper proposes a new data enhancement method that captures the similarity of sequences while ignoring the time domain characteristics of data. The proposed clustering model constructs positive and negative sample pairs by setting different shape transformation parameters,learns feature representation,and uses cross-entropy loss to maximize the similarity of positive sample pairs and minimize negative sample pairs at the instance-level and cluster-level comparison. The proposed model can jointly learn feature representation and cluster assignment in end-to-end fashion. Extensive experiments on 32 datasets in UCR show that the proposed model can obtain equal or better performance than existing methods without relying on a specific representation learning network.

Keywords　Time series clustering,Contrastive learning,Data enhancement,Representation learning,Jointly optimization

到稿日期:2022-12-06　返修日期:2023-02-28
基金项目:国家自然科学基金(62276162);山西省重点研发计划(202102070301019);山西省基础研究计划(201901D211170,202103021223464);南京市国际联合研发项目(202002021)
This work was supported by the National Natural Science Foundation of China(62276162),Key R&D Program of Shanxi Province,China(202102070301019),Basic Research Program of Shanxi Province(201901D211170,202103021223464) and Nanjing International Joint R&D Project(202002021).
通信作者:彭甫镕(pengfr@sxu.edu.cn)

1 引言

时间序列聚类是一种基本的无监督数据分析技术，被广泛应用于医学分析[1]和基因微阵列[2]等领域。通过时间序列聚类技术可以挖掘数据中的潜在结构和知识，有助于从大量的数据中提取有价值的信息[3-4]。现实中，时间序列聚类面临的主要问题在特征提取和相似性度量两个方面。时间序列聚类特征提取框架的演化大致分为3个阶段，如图1所示。传统的基于数据特征的做法，先进行数据预处理，再进行相似性度量[5-6]，这种方法无法捕获序列的上下文关系，且特征提取方法对高维数据不友好。为解决此类问题，科学家们提出了序列特征提取的方法[7-8]，该方法可以在降维的同时进行聚类工作。最典型的是基于 shapelets 的无监督学习方法[9]，该方法将高维时间序列聚类问题转换为低维特征段线性组合问题。然而，基于 shapelets 的方法需要依赖于预训练的特征库，且严重依赖于相关领域知识。随着深度聚类的发展，这类方法在聚类性能方面也逐渐趋于劣势。目前较为通用的时间序列聚类方法的思路为先学习数据表示，再进行聚类[10-12]，这类方法将工作重心放到了时间序列表示学习方面。为获取更好的时间序列特征表示，研究者们逐渐提出了结构复杂的时间序列表示学习网络[13]，然而表示网络在训练过程中需要消耗大量的计算资源，且得到的特征表示较为抽象，难以直观地定义样本相似性。随着 Transformer 模型[14]的发展以及在序列处理中的广泛应用，利用 Transformer 的编码器学习时间序列的表示[15]，并针对时间序列的高维特性对编码器做出改进[16]，应用于下游任务，取得了不错的效果。

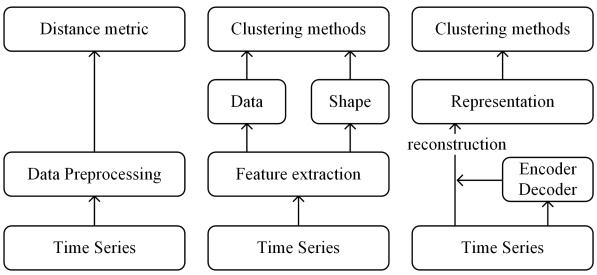

图1 时间序列特征提取框架

Fig. 1 Time series feature extraction framework

到目前为止，随着深度学习模型的发展以及各类自然语言处理和计算机视觉模型在时间序列领域的迁移应用[17-18]，现有时间序列聚类方法虽然在聚类精度上取得了不错的进展，但在以下几个方面仍然存在不足之处：

(1)基于特征提取的时间序列聚类方法一般采用两阶段分离的模式，这种方式会导致误差累计，最终只能得到次优解。

(2)基于深度学习的特征提取方法在提高特征提取质量的同时会造成模型结构复杂、相似性难以定义的问题。

(3)现有模型在面对不同领域数据集时往往需要进行针对性的调参，模型的鲁棒性不足。

为了解决上述问题，本文从时间序列的形状特征角度提出了一种数据增强方法，该方法可以捕捉序列的形状相似性，忽略数据的时域特征。同时本文使用对比学习方法从正负样本对角度定义了区间相似性，引入对比学习框架[19]，如图2所示，提出了一种不依赖于特定表示网络结构的时间序列聚类模型。通过设置不同的形状转换参数构造正负样本对学习序列的表示，并投影到特征空间中，在实例级和聚类级利用交叉熵损失最大化正样本对的相似性，最小化负样本对的相似性，实现了端到端的联合学习表示和聚类分配。

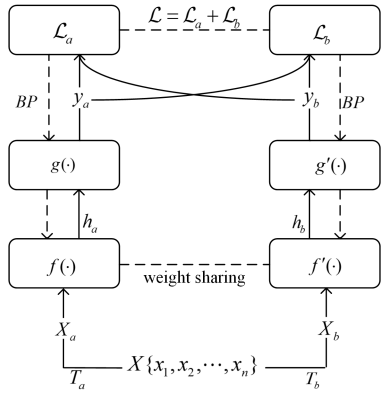

图2 对比学习框架

Fig. 2 Contrastive learning frameworks

本文的主要贡献总结如下：

(1)现有聚类方法很难直观地对时间序列相似性进行定义，严重依赖于人的经验，本文使用对比学习方法从正负样本数据角度定义了时间序列的区间相似性。

(2)提出了一种不依赖于特定表示学习网络结构的时间序列聚类模型。

(3)现有的时间序列数据增强方法难以描述时间序列的变换不变性，本文从时间序列的形状角度提出了一种适用于对比学习的数据增强方法。

(4)使用 UCR 中的32个数据集对模型进行了广泛的实验，证明了数据增强方法的有效性，且模型不依赖于复杂的深度表示网络。

2 相关工作

2.1 时间序列数据增强方法

数据增强是让有限数据样本产生更多数据价值的一种手段[20]，数据增强方法的好坏会直接影响下游任务的结果。在时间序列领域主要采用的数据增强方法包括高斯噪声、随机裁剪和序列翻转等[21-22]，这些方法普遍具有一定随机性，在数据扩充的同时会对下游任务和实验结果产生影响。在计算机视觉领域，研究者对选择有效的数据增强方法做了大量的研究，其中在图像裁剪方法中，在确保大部分正样本对语义一致的前提下，增加样本之间差异性的方法在下游任务上取得了较好的效果[23]。受该方法的启发，本文提出了一种基于形状特征的时间序列数据增强方法，对时间序列振幅进行缩放，增强了高阶特征对整体序列的影响。

2.2 时间序列聚类

随着深度聚类的发展，研究人员开始探究针对时间序列的深度聚类方法。现有的方法主要依赖于表示学习，包括学习数据的表示和使用这种表示来执行聚类任务。近年来，深度卷积神经网络(CNN)已被应用于时间序列分类任务[24-25]，

表现出具有竞争力的性能。另一种用于时间序列数据的深度聚类方法基于 RNN 模型，由于神经元之间的循环连接，该模型能够以任意的精度逼近任意的非线性系统[26]，因此可以用于捕获时间序列的动态特征。

在时间序列聚类中，为解决表示和聚类分离造成的分层训练和分段优化问题，Ghasedi Dizaji 等提出了使用联合学习框架进行时间序列端到端聚类的方法[27]。然而在联合优化中使用单一聚类损失学习深度特征表示的方法可能会破坏特征空间，学习无代表性的无意义特征，从而损害聚类性能[28]。为此，需要引入表示损失，并使用小批量随机梯度下降和反向传播同时优化表示学习网络。

在进一步的研究中发现，对表示矩阵进行交叉熵损失计算并利用余弦相似度做行列约束，可以更好地学习有利于类簇分配的特征表示。

2.3　对比学习

基于对经典孪生网络进行改进，Zagoruyko 等提出了一种基于卷积神经网络的比较学习方法[29]，并证明了这种训练方法及网络架构能够极大地提升一些机器视觉领域基础问题训练结果的准确率。随着对孪生网络的进一步研究，2020 年 Facebook 的 Chen 等指出孪生网络已成为各种无监督视觉表示学习模型的通用结构[30]。孪生网络由两个结构相同且权重共享的神经网络构成，用于判断两个输入模式的相似性，对领域知识的依赖性小，因此基于孪生网络的对比学习开始得到广泛使用[31]。对比学习着重于学习同类实例之间的共同特征，区分非同类实例之间的不同之处，基本思想是将原始数据映射到一个特征空间，其中正对的相似性最大，而负对的相似性较小[32]。与生成式学习相比，对比学习不需要关注实例上繁琐的细节，只需要在抽象语义级别的特征空间上学习对数据的区分即可，因此模型简单，泛化能力更强[33-34]。2021 年，Eldelle 等首次将对比学习的思想应用于时间序列表示学习领域[35]，提出了一种有效的时间序列的数据强弱增强方法，并使用基于时间对比模块和上下文对比模块的对比学习框架设计出一个通用的无监督时间序列表示学习模型。

综合对比学习的优点并结合对比学习方法在时间序列领域的尝试，本文采用对比学习框架训练时间序列特征提取模型，并使用实例级对比和聚类级对比对模型进行联合优化。

3　时间序列对比聚类方法

基于对比学习的时间序列聚类模型分为对比特征提取和对比约束，其中对比约束又分为实例级对比约束和聚类级对比约束两个部分。如图 3 所示，特征提取部分包括对原始时间序列进行数据增强并构建样本对，将增强后的数据映射到两个特征空间矩阵中，之后对两个特征空间矩阵做二次映射，使用实例级对比约束和聚类级对比约束来优化特征分布。本章首先引入对比特征提取，然后描述对比约束，最后介绍退化解的处理方法。

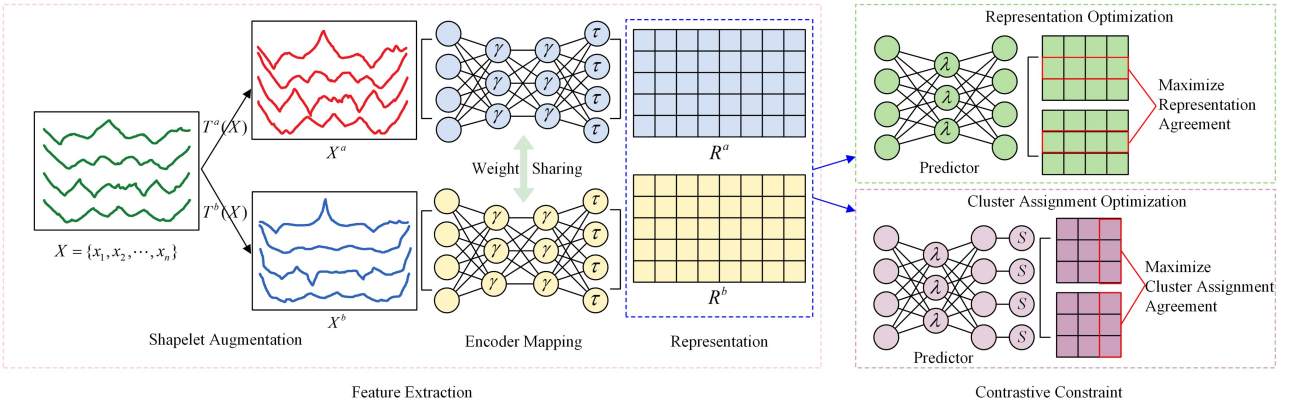

图 3　基于对比学习的时间序列聚类模型

Fig. 3　Time series clustering model based on contrastive learning

3.1　对比特征提取

对比学习的主要思想是通过比较正样本和负样本的相似性来学习样本的一般特征表示。基于对比学习的时间序列聚类方法首先对原始数据进行数据增强，同一样本的增强结果互为正样本，不同样本的增强结果互为负样本，之后使用特征编码器映射函数来学习数据的语义表示。在这一过程中数据增强方法的好坏会直接影响模型最终的性能，为此本文提出了一种新的基于形状的数据增强方法，它可以显著提高时间序列数据聚类的性能。

3.1.1　基于形状的数据增强方法

数据增强是对比学习框架在特征表示方面获得成功的主要原因之一。数据裁剪、数据翻转和高斯模糊等数据增强方法在计算机视觉领域的应用取得了成功。然而，将这些方法直接应用在时间序列聚类领域并不能取得理想的效果。将时间序列数据与图像数据进行比较可以发现，两者都属于低水平信号，并包含高级语义信息。然而，图像通过多种颜色通道和更高的维度表现出更丰富的信息。时间序列的高级语义表示更多地体现在形状方面。为此，本文提出了一种新的数据增强方法，通过时间序列数据自身的信号强度做形状扩展。对于输入的时间序列样本 $\boldsymbol{x}$ 的一种数据增强方法，将其定义为：

$$T^{\alpha}(\boldsymbol{x},a)=\boldsymbol{x}\odot\left(\frac{\boldsymbol{x}'}{\max(\boldsymbol{x}')}+\alpha\cdot\mathbf{1}\right) \tag{1}$$

$$\boldsymbol{x}'=\boldsymbol{x}-\bar{\boldsymbol{x}}\cdot 1 \tag{2}$$

其中，α 为增强系数，$\odot$ 为阿达玛乘积（Hadamard product），$\mathbf{1}$ 是只由元素 1 组成的向量，x' 是输入序列的中心元素，$\bar{x}$ 是输入序列的平均值。由式(1) 可以得出，数据增强的最终结果与 $\boldsymbol{x}_i$ 的大小成正相关关系，引入增强系数 α 可以控制数据的

缩放比例。在对比学习方法中，需要对样本的两种表现形式进行相似性度量，通过调整式(1)中的增强系数 α 可以获得两种不同的数据增强。在实验过程中发现，使用二次增强的方法可以得到更好的聚类结果，因此设置其中一次增强为：

$$T^b(\boldsymbol{x},\beta)=T^a(T^a(\boldsymbol{x},\beta),\beta) \tag{3}$$

使用式(1)和式(3)中所示的两种方法进行数据增强，可以得到相对稳定的数据增强结果。通过设置参数 $\alpha=2$ 和 $\beta=0.5$ 进行数据增强，可以得到较好的实验结果。

3.1.2 序列编码器

对于一般深度学习模型，编码器的选择至关重要，它可以直接影响从数据中提取到的特征的好坏，大多数时间序列聚类模型都需要选择特定的编码器函数进行样本表示学习。本文采用对比学习方法提出了一种不依赖于特定表示学习网络结构的时间序列聚类框架。本文以多层感知机(MLP)为例来解释模型的工作原理。

将数据增强得到的两组样本按行进行排列得到两个增强矩阵 $\boldsymbol{X}^t$，其中 $t\in(a,b)$ 分别代表不同的增强方式，表示学习的编码器网络可以写为：

$$\boldsymbol{R}^t=h(\boldsymbol{X}^t)=\boldsymbol{W}_3^h\boldsymbol{o}_2^h+\boldsymbol{b}_3^h \tag{4}$$

$$\boldsymbol{O}_2^h=\mathrm{Relu}(\boldsymbol{W}_2^h\boldsymbol{o}_1^h+\boldsymbol{b}_2^h) \tag{5}$$

$$\boldsymbol{O}_1^h=\mathrm{Relu}(\boldsymbol{W}_1^h\boldsymbol{X}_1^t+\boldsymbol{b}_1^h),t\in\{a,b\} \tag{6}$$

使用 MLP_3 表示编码器网络，使用 Relu 作为激活函数。在实际应用中可以根据不同数据集，使用 RNN，LSTM 或者 CNN 代替 MLP_3 作为编码器网络。本文通过实验证明了这些编码器在大量数据集上有着较好的聚类性能。

3.2 对比约束

在表示学习和聚类过程中，本文将数据划分成大小为 N 的 batch，使用单个 batch 对模型进行多次训练。在对比特征提取阶段，首先使用数据增强方法将 N 个数据扩充成为 N 个二元组，每个二元组由同一样本经过两种数据增强方法后得到，每个二元组内的样本互为正样本，不同元组内的样本互为负样本。依据这一规则，对于同一个 batch，每个样本有 1 个正样本和 $2N-2$ 个负样本，将这些样本依据增强方法的不同分配到两个增强矩阵中，最后通过序列编码器得到两个特征表示矩阵。本文使用实例级对比约束和聚类级对比约束共同作用于序列编码器，以同时优化表示和聚类性能。

3.2.1 实例级对比约束

直接对表示矩阵 $\{\boldsymbol{R}^a,\boldsymbol{R}^b\}$ 进行对比约束可能会导致解退化，因此本文使用一个两层的 MLP 函数 $g(\cdot)$ 将表示矩阵映射到一个新的空间中，对映射后的矩阵进行表示约束。

$$\boldsymbol{Z}^t=g(\boldsymbol{R}^t)=\boldsymbol{W}_2^g\boldsymbol{O}_1^g+\boldsymbol{b}_1^g \tag{7}$$

$$\boldsymbol{O}_1^g=\mathrm{Relu}(\boldsymbol{W}_2^g\boldsymbol{R}^t+\boldsymbol{b}_1^g),t\in\{a,b\} \tag{8}$$

其中，$\boldsymbol{W}_2^g$，$\boldsymbol{W}_1^g$ 和 $\boldsymbol{b}_2^g$，$\boldsymbol{b}_1^g$ 分别为特征映射网络 $g(\cdot)$ 第二层和第一层的权重和偏置参数。在新的特征空间中设映射后的特征表示矩阵为 $\boldsymbol{Z}=[\boldsymbol{Z}^a,\boldsymbol{Z}^b]$，使用 $s(\cdot,\cdot)$ 表示余弦相似度对 $\boldsymbol{Z}$ 中两个矩阵进行相似度度量：

$$s(u,v)=\frac{\langle u,v\rangle}{\|u\|\cdot\|v\|} \tag{9}$$

其中，$\langle\cdot,\cdot\rangle$ 表示内积运算，$\|\cdot\|$ 表示向量的二范数，给定任意一个表示向量 $\boldsymbol{z}_i$，通过计算余弦相似度最大化与正样本之间相似性，并最小化 $\boldsymbol{z}_i$ 与其余 $2N-2$ 个负样本之间的相似性，该过程可以通过式(10)来实现。

$$l_i^r=-\log\frac{\exp(s(\boldsymbol{z}_i,\boldsymbol{z}_i')/\tau)}{\sum_{k=1}^{2N}1_{[k\neq i,k\neq i']}\exp(s(\boldsymbol{z}_i,\boldsymbol{z}_i')/\tau)} \tag{10}$$

此处，$1_{[k\neq i,k\neq i']}\in\{0,1\}$ 为指示函数，τ 为温度参数，通过式(10)对 $\boldsymbol{Z}=[\boldsymbol{Z}^a,\boldsymbol{Z}^b]$ 的行和列分别进行余弦相似度计算，得到实例级对比约束的损失为：

$$\mathscr{L}^r=\frac{1}{2N}\sum_{i=1}^{2N}l_i^r \tag{11}$$

3.2.2 聚类级对比约束

对于一般的编码器网络，使用式(11)即可完成特征表示矩阵的学习和优化。许多时间序列聚类工作在得到样本的特征表示以后，再使用聚类方法获得最终的聚类结果。然而这种做法无疑会造成表示优化和聚类优化的分离，在增大了工作量的同时还无法避免两阶段优化造成的误差。为此，本文遵循对比聚类的思想，在对特征表示矩阵的学习过程中引入了聚类级对比约束，在进一步优化特征表示的同时还可以端到端地学习聚类分配。此方法的基本思想为，同一组数据进行两次增强后得到的聚类结果中类簇分配一致。通过序列编码器网络学习到的两个表示矩阵 $\{\boldsymbol{R}^a,\boldsymbol{R}^b\}$，对应的行与行、列与列上的特征表示应尽可能相似。本文使用多层感知分类器 $f(\cdot)$ 来度量特征表示的相似性，并以概率的形式进行展示。

$$\boldsymbol{Y}^t=f(\boldsymbol{R}^t)=\mathrm{softmax}(\boldsymbol{W}_2^f\boldsymbol{O}_1^f+\boldsymbol{b}_2^f) \tag{12}$$

$$\boldsymbol{O}_1^f=\mathrm{Relu}(\boldsymbol{W}_1^f\boldsymbol{Z}^t+\boldsymbol{b}_1^f),t\in\{a,b\} \tag{13}$$

其中，$\boldsymbol{Y}^a,\boldsymbol{Y}^b\in\mathbb{R}^{N\times K}$ 为类别分配，K 为类别数量。每一行 $\boldsymbol{Y}^a$ 和 $\boldsymbol{Y}^b$ 中元素的累加和为 1。在聚类约束中，$\boldsymbol{Y}^a$ 和 $\boldsymbol{Y}^b$ 的相同行元素应该尽可能相似，不同行元素应该尽可能不同。令 $\boldsymbol{Y}'=[\boldsymbol{Y}^a,\boldsymbol{Y}^b]\in\mathbb{R}^{2K\times N}$，可以使用式(14)进行聚类级对比约束：

$$l_i^c=-\log\frac{\exp(s(y_i',y_{i'}')/\tau)}{\sum_{k=1}^{2K}1_{[k\neq i,k\neq i']}\exp(s(y_i',y_k'))} \tag{14}$$

其中，i 和 i' 为 $\boldsymbol{Y}^a$ 和 $\boldsymbol{Y}^b$ 中同类的样本，l_i^c 可以在最大化相同样本之间聚类表示的同时降低不同样本之间的表示。在 l_i^c 基础上可以得到聚类级对比约束的损失函数为：

$$\mathscr{L}^c=\frac{1}{2K}\sum_{k=1}^{2K}l_k^c \tag{15}$$

基于对比学习的时间序列聚类方法具体描述如算法 1 所示。

算法 1 基于对比学习的时间序列聚类方法

输入：数据集 $\mathscr{D}$；类簇数量 K；迭代次数 T；batch 大小 N；数据增强参数 α，β；约束参数 τ_1，τ_2

输出：聚类结果矩阵 $\mathbf{Y}$

```
1. //训练
2. for iter=1 to T do
3.     划分 batch：X={x_i}_{i=1}^N ∈ D
4.     使用式(1)进行数据增强：X^a=T^a(X,α)
5.     使用式(2)进行数据增强：X^b=T^b(X,β)
6.     根据式(4)－式(6)计算表示：
       R^a=h(X^a)，R^b=h(X^b)
7.     根据式(7)进行实例级映射：
```

$\mathbf{Z}_a=g(\mathbf{R}^a)$，$\mathbf{Z}_b=g(\mathbf{R}^b)$

8. $\mathbf{Z}\equiv[\mathbf{Z}^a,\mathbf{Z}^b]$

9. 根据式(12)进行类别预测：

$\mathbf{Y}_a=f(\mathbf{R}^a)$，$\mathbf{Y}_b=f(\mathbf{R}^b)$

10. $\mathbf{Y}'\equiv[\mathbf{Y}^a,\mathbf{Y}^b]$

11. 使用式(11)计算实例级对比约束损失$\mathcal{L}^r$

12. 使用式(14)计算聚类级对比约束损失$\mathcal{L}^c$

13. 使用式(17)计算聚类正则化损失$\mathcal{L}^{reg}$

14. 计算整体损失：$\mathcal{L}=\mathcal{L}^r+\mathcal{L}^c+\mathcal{L}^{reg}$

15. 使用 Adam 最小化 $\mathcal{L}$ 更新网络 h(·)，g(·)，f(·)

16. end

17. //预测

18. 类别预测矩阵：$\mathbf{Y}=f(h(\mathcal{D}))$

19. 返回 $\mathbf{Y}$

3.3　损失函数

在实验过程中发现，若仅使用实例级对比约束和聚类级对比约束作为模型的损失函数，会产生将大量样本归为同一类的情况。为此，本文在模型训练过程中引入了最大化概率分配熵，以解决跨类别样本分布不均衡的问题。对于任意类别 k，平均分配概率可以近似为：

$$P_k^t=\frac{1}{N}\sum_{i=1}^{N}Y_{i,k}^t,t\in\{a,b\} \tag{16}$$

通过最大化所有类别分配概率的熵对类簇分配进行正则化，表示为：

$$\mathcal{L}^{reg}=-\frac{1}{K}\sum_{k=1}^{K}[P_k^a\log(P_k^a)+P_k^b\log(P_k^b)] \tag{17}$$

模型的总体损失函数可以由实例级对比损失、聚类级对比损失、类簇分配正则化 3 个部分的线性组合来表示。实验中设置线性组合参数均为 1 即可取得较好的聚类效果。

$$\mathcal{L}=\mathcal{L}^r+\mathcal{L}^c+\mathcal{L}^{reg} \tag{18}$$

通过最小化总体损失函数可以优化表示网络 $h(\cdot)$和预测网络 $f(\cdot)$。许多时间序列聚类算法需要训练一个表示网络和一个聚类算法，例如 SimCLR[36] 和 DTCR[11]，然而本文提出的模型可以将这两个步骤合并输出类别结果，并通过实例级和聚类级两种不同约束对整体损失进行联合优化。最终的聚类结果 $\boldsymbol{Y}$ 可以表示为：

$$\boldsymbol{Y}=f(g(\boldsymbol{X})) \tag{19}$$

其中，$\boldsymbol{Y}$ 为每行单独使用 one-hot 编码的结果矩阵，每行代表一个样本的聚类结果。

4　实验

4.1　数据集及评价指标

为了评估模型的有效性，本文在 32 个来自不同领域的 UCR 数据集上进行实验，以验证模型的有效性。UCR 中的每个数据集都包含一个 TRAIN.ts 和 TEST.ts 文件用于训练和测试，实验设置将两个文件合并用于聚类。数据集信息如表 1 所列。

表 1　数据集及其数据描述

Table 1　Dataset and their data descriptions

Dataset	Sample	Length	Class	Dataset	Sample	Length	Class
ACSF1	200	1 460	10	InsectWingbeatSound	2 200	256	11
ArrowHead	211	251	3	Mallat	2 400	1 024	8
Coffee	56	286	2	MedicalImages	1 141	99	10
DistalPhalanxOutlineAgeGroup	539	80	3	MiddlePhalanxOutlineAgeGroup	554	80	3
DistalPhalanxTW	539	80	6	MiddlePhalanxTW	553	80	6
ECG200	200	96	2	MixedShapesSmallTrain	2 525	1 024	5
ECG5000	5 000	140	5	MoteStrain	1 272	84	2
FaceFour	112	350	4	Plane	210	144	7
FreezerRegularTrain	3 000	301	2	PowerCons	360	144	2
FreezerSmallTrain	3 000	301	2	ProximalPhalanxOutlineAgGroup	605	80	3
GunPointOldVersusYoung	451	150	2	ProximalPhalanxTW	605	80	6
HandOutlines	1 370	2 709	2	Rock	70	2 844	4
Haptics	463	1 092	5	SemgHandMovementCh2	900	1 500	6
InlineSkate	650	1 882	7	SemgHandSubjectCh2	900	1 500	5
InsectEPGRegularTrain	311	601	3	ShapesAll	1 200	512	60
InsectEPGSmallTrain	311	601	3	Symbols	1 020	398	6

实验设置模型的整体学习率为 0.001，模型使用 Adam 优化器进行优化，其中权值衰减参数设置为 0.001，表示学习网络使用一个三层的 MLP，设置隐藏层的大小为 128，输出层的大小为 256，前两层使用 Relu 作为激活函数，第三层使用 Tanh 作为激活函数。在对比约束模块，实例级对比约束的温度参数设置为 0.5，映射矩阵的列空间维度为 128，聚类级对比约束中设置温度参数为 1.0，输出的类簇分配概率矩阵列空间维度等于类别数。在此参数设置下，模型可以取得较好的聚类结果。实验使用 Intel(R) Xeon(R) CPU E5-2650 v4 @ 2.20GHz CPU，Tesla P100 PCIe 16GB GPU。在表 1 所列的 32 个时间序列数据集上进行 500 轮迭代，共耗时 3h。

为了直观地体现所提模型的性能，本文使用了 3 种广泛使用的聚类指标[37]来评估模型，包括标准互信息(NMI)、调整兰德指数(ARI)和准确率(ACC)。

4.2　聚类性能实验

为验证模型的有效性，本文选用 DEC[38]，DEPICT[27]，IDEC[28]，DTCR[11]，SDCN[39]作为对比实验模型。具体描述如下。

DEC：提出了一种基于软聚类任务的辅助目标分类方法来迭代和细化聚类过程，这一过程可以同时改进聚类和特征表示，联合解决特征空间学习和类簇关系判别。

DEPTIC：模型的表示网络由一个堆叠在多层卷积自编

码器之上的多项式逻辑回归函数组成，使用相对熵最小化方法定义一个聚类目标函数，并使用聚类分配的先验概率进行正则化。引入了一个联合学习框架，对聚类和重构的损失函数进行联合优化，并更新所有网络层参数。

IDEC：使用一个欠完全自动编码器约束数据生成分布的局部结构，并通过整合聚类损失和自编码器的重构损失优化特征空间的学习，并验证该优化问题可以通过小批量随机梯度下降和反向传播得到有效的解决。

DTCR：使用假样本生成策略并引入辅助分类任务进行数据扩充，以提高编码器的编码能力，并将时间重建和 K-means 聚类目标集成到 seq2seq 特征提取模型中，优化聚类分配。

SDCN：设计了一个传递算子，将自动编码器学习到的表示转移到相应的 GCN 层，并设计了一个双重自监督机制来统一这两种不同的深度神经结构，将结构性信息应用到了深度聚类中。

聚类性能对比实验结果[1)]如表 2 所列。实验结果表明，在表 1 所列的 32 个数据集上本文所提模型的聚类性能均与对比方法相当或优于对比方法。为了直观地展示聚类过程中 NMI，ARI，ACC 这 3 个评价指标对应的聚类性能变化情况，本文以 Mallat 数据集为例对迭代过程中聚类性能与迭代次数关系进行可视化，如图 4 所示。

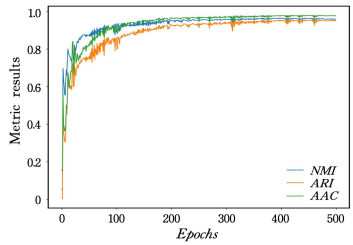

图 4 聚类结果与迭代次数的关系

Fig. 4 Relation between clustering results and epochs

此外，为了更直观地展示模型的训练过程，本文以 Mallat 数据集为例使用 t-SNE 降维方法[40]对类簇演化过程进行可视化。实验在标准参数设置情况下进行 100 轮迭代训练，每执行 20 轮迭代训练使用现有模型进行一次聚类标签预测。为了验证序列编码器在迭代过程中的优化情况，首先对原始时间序列做数据增强，使用当前阶段的序列编码器进行编码，得到对应迭代次数优化下的时间序列特征表示，最后使用 t-SNE 进行数据降维并在二维坐标系上进行结果展示。实验结果如图 5 所示，从可视化结果可以看出模型在迭代优化过程中类簇变化的情况，整体表现为类内相似度高，类间相似度低，符合聚类的基本思想。

表 2 32 个时间序列数据集上聚类结果比较

Table 2 Comparison of clustering results on 32 time series datasets

Method	DEC			DEPTIC			IDEC			DTCR			SDCN			Ours		
Metrics	*NMI*	*ARI*	*ACC*	*NMI*	*ARI*	*ACC*	*NMI*	*ARI*	*ACC*	*NMI*	*ARI*	*ACC*	*NMI*	*ARI*	*ACC*	*NMI*	*ARI*	*ACC*
ACSF1	0.560	0.283	0.460	0.523	0.229	0.450	**0.592**	0.318	0.500	0.477	0.166	0.390	0.560	0.262	0.480	0.573	**0.364**	**0.555**
ArrowHead	0.236	0.264	0.640	**0.389**	**0.365**	**0.703**	0.298	0.235	0.571	0.231	0.226	0.611	0.232	0.238	0.629	0.333	0.321	0.668
Coffee	0.695	0.725	0.929	**1.000**	**1.000**	**1.000**	0.812	0.857	0.964	**1.000**	**1.000**	**1.000**	**1.000**	**1.000**	**1.000**	**1.000**	**1.000**	**1.000**
DistalPhalanxOutlineAgeGroup	0.340	0.265	0.698	0.308	0.239	0.705	0.369	0.308	0.741	0.320	0.292	0.719	0.413	0.332	0.748	**0.459**	**0.463**	**0.807**
DistalPhalanxTW	0.616	0.513	0.662	0.535	0.482	0.669	0.602	0.502	0.691	0.522	0.507	0.626	0.554	0.529	0.626	**0.629**	**0.675**	**0.750**
ECG200	0.155	0.237	0.670	0.054	0.080	**0.900**	0.136	0.214	0.700	0.119	0.192	0.730	0.040	0.079	0.830	0.118	0.203	0.765
ECG5000	0.527	0.600	0.752	0.564	0.733	0.801	0.472	0.616	0.840	0.532	0.602	0.801	0.555	0.614	0.697	**0.591**	**0.760**	**0.883**
FaceFour	0.527	0.448	0.739	0.599	0.587	**0.795**	0.572	0.523	0.716	**0.691**	**0.602**	0.739	0.570	0.515	0.761	0.487	0.406	0.661
FreezerRegularTrain	0.219	0.287	0.768	0.221	0.289	0.769	0.221	0.287	0.768	0.263	**0.335**	**0.789**	0.227	0.296	0.772	**0.271**	0.281	0.765
FreezerSmallTrain	0.229	0.289	0.769	0.226	0.292	0.770	0.237	0.309	0.778	**0.254**	**0.310**	**0.779**	0.232	0.303	0.775	0.220	0.284	0.767
GunPointOldVersusYoung	0.142	0.188	0.718	0.427	**0.509**	**0.857**	0.369	0.464	0.841	**0.431**	0.491	0.851	0.247	0.321	0.784	0.237	0.308	0.778
HandOutlines	0.258	0.332	0.789	**0.351**	**0.473**	**0.846**	0.330	0.449	0.838	0.242	0.341	0.797	0.324	0.442	0.835	0.160	0.211	0.730
Haptics	0.110	0.073	0.367	0.102	0.082	0.380	**0.116**	0.073	**0.390**	0.097	0.058	0.380	0.097	0.052	0.360	**0.116**	**0.091**	0.374
InlineSkate	0.075	0.032	0.244	**0.112**	**0.039**	**0.280**	0.063	0.033	0.253	0.098	0.037	0.258	0.068	0.031	0.253	0.047	0.015	0.214
InsectEPGRegularTrain	0.247	0.281	0.647	0.244	0.234	0.663	0.308	0.299	0.687	0.358	0.356	0.707	0.194	0.194	0.614	**1.000**	**1.000**	**1.000**
InsectEPGSmallTrain	0.219	0.230	0.614	0.239	0.291	0.655	0.247	0.230	0.622	0.333	0.361	0.699	0.309	0.383	0.731	**1.000**	**1.000**	**1.000**
InsectWingbeatSound	0.545	0.366	0.505	0.493	0.334	0.513	0.508	0.346	0.527	0.529	0.356	0.560	**0.569**	**0.385**	**0.572**	0.459	0.286	0.473
Mallat	0.904	0.780	0.816	0.870	0.739	0.797	0.900	0.768	0.798	0.796	0.697	0.817	0.955	0.947	0.975	**0.969**	**0.963**	**0.983**
MedicalImages	0.210	**0.180**	**0.470**	0.111	0.039	0.386	0.227	0.102	0.347	**0.316**	0.075	0.334	0.295	0.106	0.336	0.208	0.068	0.289
MiddlePhalanxOutlineAgeGroup	0.139	0.082	0.617	0.159	0.147	0.649	0.143	0.158	0.643	0.017	0.003	0.565	0.116	0.066	0.526	**0.421**	**0.440**	**0.751**
MiddlePhalanxTW	**0.485**	0.320	0.578	0.472	0.367	**0.591**	0.465	0.378	0.578	0.435	0.424	0.539	0.433	0.418	0.539	0.466	**0.510**	**0.579**
MixedShapesSmallTrain	0.483	0.471	0.686	0.530	0.499	0.669	0.499	0.478	0.691	**0.598**	**0.561**	0.706	0.510	0.492	**0.719**	0.494	0.223	0.441
MoteStrain	0.401	0.496	0.852	**0.444**	**0.518**	**0.860**	0.435	0.500	0.854	0.044	0.064	0.627	0.402	0.476	0.845	0.049	0.071	0.634
Plane	0.912	0.872	0.943	0.931	0.896	0.952	0.933	0.914	0.962	0.927	0.871	0.876	**0.963**	**0.956**	**0.981**	0.898	0.829	0.862
PowerCons	0.399	0.487	0.850	0.391	0.487	0.850	**0.493**	**0.585**	**0.883**	0.298	0.370	0.806	0.416	0.503	0.856	0.162	0.064	0.628
ProximalPhalanxOutlineAgeGroup	0.560	**0.661**	**0.868**	**0.573**	0.637	0.854	**0.573**	0.637	0.854	0.532	0.596	0.805	0.523	0.638	0.854	0.563	0.570	0.779
ProximalPhalanxTW	**0.703**	0.721	**0.824**	0.596	0.562	0.722	0.643	0.663	0.780	0.638	0.711	0.766	0.614	0.604	0.654	0.632	**0.738**	0.775
Rock	**0.453**	0.272	0.580	0.247	0.171	0.560	0.386	0.172	0.540	0.316	0.208	0.560	0.451	0.277	0.580	0.440	**0.332**	**0.629**
SemgHandMovementCh2	**0.221**	0.123	0.364	0.153	0.086	0.371	0.199	0.114	0.371	0.188	0.093	0.344	0.181	0.104	0.356	0.213	**0.139**	**0.387**
SemgHandSubjectCh2	**0.392**	**0.319**	0.536	0.378	0.306	**0.549**	0.391	0.292	0.547	0.200	0.121	0.447	0.285	0.216	0.502	0.314	0.266	0.478
ShapesAll	0.687	0.295	0.462	**0.737**	0.374	0.518	0.719	0.340	0.502	0.716	0.335	0.512	0.734	0.369	0.512	0.697	**0.388**	**0.523**
Symbols	0.840	0.763	0.878	0.825	0.738	0.860	**0.875**	**0.832**	**0.918**	0.794	0.660	0.745	0.811	0.698	0.749	0.790	0.655	0.675
# Average	0.422	0.383	0.665	0.431	0.401	**0.686**	0.442	0.406	0.678	0.416	0.376	0.653	0.434	0.401	0.670	**0.469**	**0.435**	0.675
# Variance	**0.056**	**0.050**	**0.031**	0.064	0.062	0.034	**0.056**	0.056	0.034	0.066	0.064	0.032	0.067	0.067	0.037	0.085	0.090	0.042
# Best	6	4	3	7	6	10	5	2	3	6	5	3	3	3	4	**10**	**15**	**12**
# Total	13			23			10			14			10			**37**		

1) https://github.com/yangbo981205/CTSC

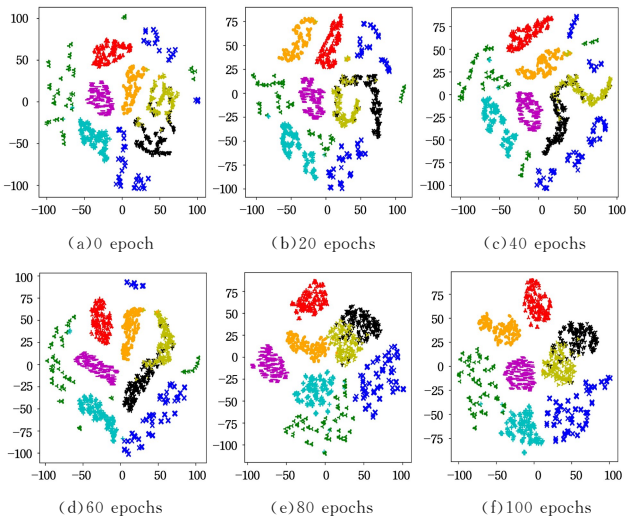

图 5　迭代过程中类簇演化

Fig. 5　Evolution of class clusters during iteration

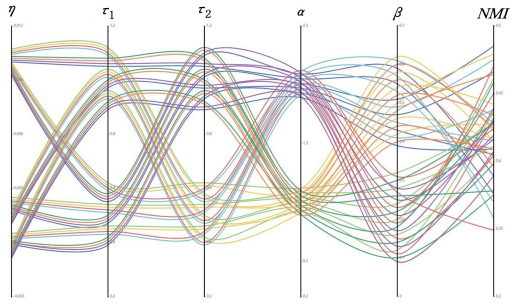

图 6　网格化调参实验结果

Fig. 6　Results of grid search experiments

此外，现有模型在面对不同领域数据集时往往需要进行针对性的调参，模型的鲁棒性不足。本文模型可以在一组相同的参数设置下对不同数据集表现出较优的聚类性能。在实验过程中，为了确定该组实验参数，本文使用网格化调参方法对模型中的主要参数进行了评估与改进，包括数据增强参数 α，β，约束参数 τ_1，τ_2，以及学习率 η。在前期的单独参数实验中可以发现，将每个参数约束在一个对应的集合上可以取得较好的实验结果，因此设置 $\eta\in\{0.01,0.001,0.0001\}$，$\alpha\in\{1,2\}$，$\beta\in\{0.2,0.5\}$，$\tau_1\in\{0.5,1.0\}$，$\tau_2\in\{0.5,1.0\}$，使用网格化调参方法可以得到 48 种参数组合形式。为了节约模型训练时间，在调参阶段将迭代次数固定为 100，正式训练阶段的迭代次数为 500。实验结果为在表 1 所列的 32 个时间序列数据集上，由 5 个参数变量控制得到的 NMI 指标的评价结果的算术平均值。使用 Parallel Coordinates 对实验结果进行可视化分析，为了使可视化结果对于不同的参数组合具有较好的区分性，在作图阶段对 η 的万分位进行近似表示，对 α，β，τ_1，τ_2 的百分位进行近似表示，读图时需要根据参数区间对坐标轴进行近似读取。实验结果如图 6 所示。对图示聚类指标 NMI 所示结果进行回溯发现，聚类性能表现较好的参数组中的各参数取值较为相近，并得出结论：当参数取值 $\eta=0.001$，$\tau_1=0.5$，$\tau_2=1$，$\alpha=2$，$\beta=0.5$ 时聚类性能表现最佳，因此本文选择该组参数作为模型训练参数。

4.3　数据增强方法实验

为解决现有时间序列数据增强方法与对比学习框架不匹配的问题，本文从时间序列的形状角度提出了一种新的数据增强方法。相比原始时间序列数据和高斯噪声、序列翻转、序列裁剪等传统的数据增强，使用该方法可以在聚类任务上达到优于现有数据增强方法的聚类结果。为了验证该方法的有效性，实验设置对表 1 所列的 32 个数据集进行增强，并使用 MLP 作为序列编码器，在对比学习框架中进行聚类实验，选取 8 组具有代表性的结果进行展示，实验结果如图 7 所示。

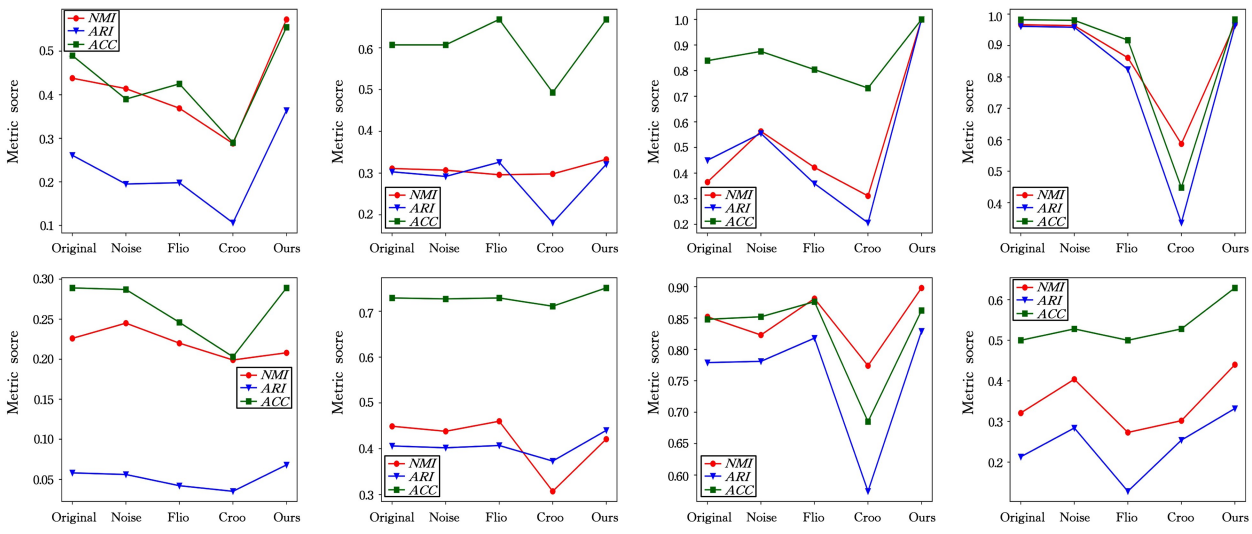

图 7　不同数据增强方法的对比实验结果

Fig. 7　Experimental result comparison of different data augmentation methods

从实验结果可以看出，相比使用原始数据进行对比学习以及使用高斯噪声、序列翻转、序列裁剪等传统的数据增强方法增强后的数据进行对比学习，本文提出的基于形状的数据增强方法均体现了出明显的优势。究其原因，基于形状的数据增强可以通过序列自身的信号强弱进行形状扩展，扩展结果如图 8 所示，在对比学习过程中使用增强后的两个数据

进行形状相似性学习，可以定义出样本的区间相似性，增强对比学习在时间序列聚类方面的泛化性能。

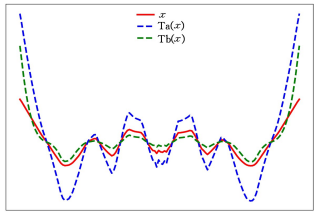

图 8 数据增强结果

Fig. 8 Data enhancement results

4.4 表示学习网络结构实验

现有时间序列聚类模型的聚类性能严重依赖于使用复杂的表示学习网络提取数据特征，而所提模型不依赖于复杂的表示网络。为了验证这一特性，实验分别设置使用 MLP，LSTM，CNN，RNN 这 4 种网络作为表示网络进行序列编码，统一使用基于形状的数据增强方法做数据增强，在实例级对比中将映射输出维度统一设置为 128，温度系数设置为 0.5，聚类级对比温度系数设置为 1.0。在对不同的表示网络进行针对性调参后得到的实验结果如图 9 所示。使用参数表示学习网络参数设置如表 3 所列。

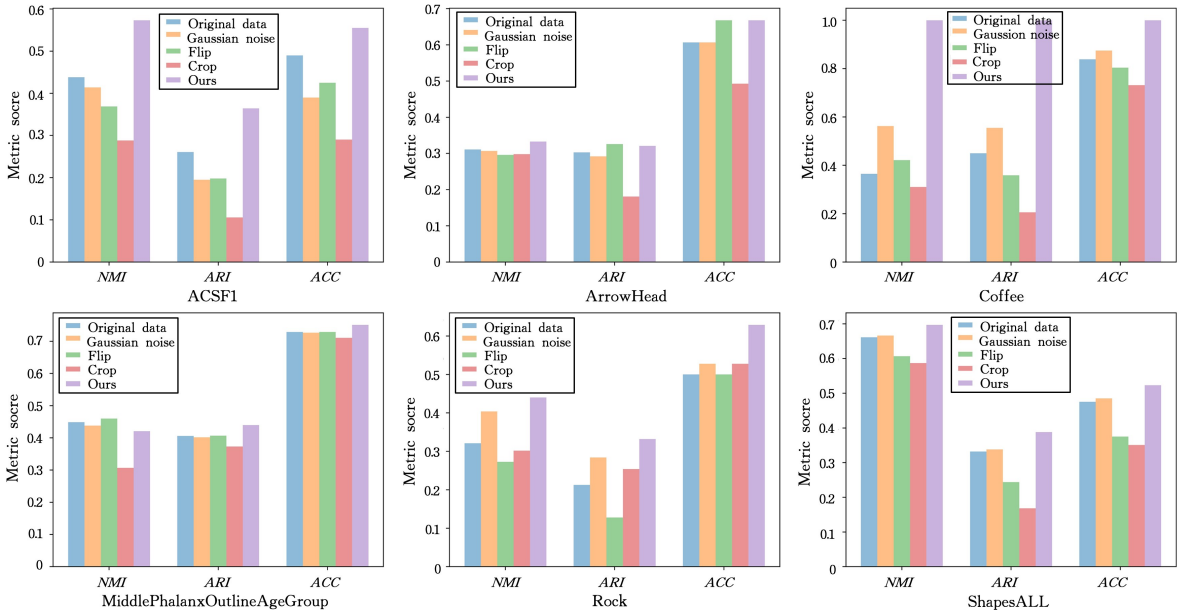

图 9 不同表示网络的聚类性能

Fig. 9 Clustering performance of different representation learning networks

表 3 不同表示学习网络的参数设置

Table 3 Parameter settings of different representation learning networks

Network	Parameter
MLP	*hidden_size*=128 *representation_dim*=256
LSTM	*hidden_size*=1 024 *representation_dim*=512 *num_layers*=3
CNN	*representation_dim*=512 *kernel_size*=3
RNN	*hidden_size*=1 024 *representation_dim*=512 *num_layers*=3

其中，*hidden_size* 表示网络隐藏层大小，*representation_dim* 表示网络最终的输出维度，*num_layers* 表示网络的层数，*kernel_size* 表示 *CNN* 网络卷积核的大小。

从实验结果可以看出，针对不同类型的数据集，4 种表示学习网络得到的结果相似且都有各自的优势，足以证明表示学习框架在解决时间序列聚类问题方面不依赖于特定的表示学习网络。

结束语 本文基于对比学习方法从正负样本数据的角度定义了时间序列的区间相似性，提出了一种适用于时间序列数据聚类任务的对比学习模型，同时发现现有时间序列数据增强方法难以描述时间序列的变换不变性。为此，本文提出了一种新的基于时间序列形状特征的数据增强方法，该方法可以很好地适用于对比学习框架。通过设置实例级对比头和聚类级对比头并引入类簇分配正则化联合优化表示网络学习。最后，通过设置聚类性能对比实验、数据增强方法对比实验，以及表示学习网络对比实验，验证所提方法的有效性。

在实验过程中发现，本文提出的基于对比学习的时间序列聚类框架虽然不依赖于特定的表示学习网络，但是对于网络和数据集的适配度问题，在现有实验中还不能得到证实。在后续的工作中会重点研究不同表示学习网络和不同数据集特征的匹配性问题。

参 考 文 献

[1] HIRANO S, TSUMOTO S. Cluster analysis of time-series medical data based on the trajectory representation and multiscale comparison techniques[C]// Sixth International Conference on Data Mining(ICDM'06). IEEE, 2006: 896-901.

[2] YIN Y, ZHAO Y H, ZHANG B, et al. Clustering of synchro-

nous and asynchronous co-regulated genes in time-series microarray data[J]. Journal of Computer Science,2007(8):1302-1314.

[3] AGHABOZORGI S,SHIRKHORSHIDI A S,WAH T Y. Time-series clustering — a decade review[J]. Information Systems,2015,53:16-38.

[4] LI H L,ZHANG L P. A Survey of Clustering Research in Time Series Data Mining[J]. Journal of University of Electronic Science and Technology of China,2022,51(3):416-424.

[5] NELSON B K. Time series analysis using autoregressive integrated moving average(ARIMA) models[J]. Academic Emergency Medicine,1998,5(7):739-744.

[6] CHEN H Y,LIU C H,SUN B. A review of similarity measures for time series data mining[J]. Journal of Control and Decision,2017,32(1):1-11.

[7] YE L,KEOGH E. Time series shapelets:a new primitive for data mining[C]// Proceedings of the 15th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. 2009:947-956.

[8] REN S G,ZHANG J X,GU X J,et al. A Review of Research on Time Series Feature Extraction Methods[J]. Journal of Chinese Computer Systems,2021,42(2):271-278.

[9] ZHANG Q,WU J,ZHANG P,et al. Salient subsequence learning for time series clustering[J]. IEEE Transactions on Pattern Analysis and Machine Intelligence,2018,41(9):2193-2207.

[10] FORTUIN V,HüSER M,LOCATELLO F,et al. Som-vae:Interpretable discrete representation learning on time series[C]// International Conference on Learning Representations. 2019.

[11] MA Q,ZHENG J,LI S,et al. Learning representations for time series clustering[J]. Advances in Neural Information Processing Systems,2019,32:3776-3786.

[12] XU Y X,ZHAO J F,WANG Y S,et al. Time-series knowledge graph representation learning[J]. Computer Science, 2022,49(9):162-171.

[13] LAFABREGUE B,WEBER J,GANÇARSKI P,et al. End-to-end deep representation learning for time series clustering:a comparative study[J]. Data Mining and Knowledge Discovery,2022,36(1):29-81.

[14] VASWANI A,SHAZEER N,PARMAR N,et al. Attention is all you need[C]// Proceedings of the 31st International Conference on Neural Information Processing Systems. 2017:6000-6010.

[15] ZERVEAS G,JAYARAMAN S,PATEL D,et al. A transformer-based framework for multivariate time series representation learning[C]// Proceedings of the 27th ACM SIGKDD Conference on Knowledge Discovery & Data Mining. 2021:2114-2124.

[16] ZHOU H,ZHANG S,PENG J,et al. Informer:Beyond efficient transformer for long sequence time-series forecasting[C]// Proceedings of the AAAI Conference on Artificial Intelligence. 2021,35(12):11106-11115.

[17] HE H,ZHANG Q,BAI S,et al. CATN:Cross Attentive Tree-aware Network for Multivariate Time Series Forecasting[C]// Proceedings of the AAAI Conference on Artificial Intelligence. 2022,36(4):4030-4038.

[18] WOO G,LIU C,SAHOO D,et al. CoST:Contrastive Learning of Disentangled Seasonal-Trend Representations for Time Series Forecasting[C]// International Conference on Learning Representations. 2022.

[19] LI Y,HU P,LIU Z,et al. Contrastive clustering[C]// Proceedings of the AAAI Conference on Artificial Intelligence. 2021:8547-8555.

[20] SHORTEN C,KHOSHGOFTAAR T M. A survey on image data augmentation for deep learning[J]. Journal of Big Data,2019,6(1):1-48.

[21] YUAN J D,WANG Z H. A Survey of Time Series Representation and Classification Algorithms[J]. Computer Science,2015,42(3):1-7.

[22] WEN Q,SUN L,YANG F,et al. Time series data augmentation for deep learning:A survey[C]// International Joint Conference on Artificial Intelligence. IJCAI,2021:4653-4660.

[23] PENG X,WANG K,ZHU Z,et al. Crafting better contrastive views for siamese representation learning[C]// Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2022:16031-16040.

[24] CUI Z,CHEN W,CHEN Y. Multi-scale convolutional neural networks for time series classification[J]. arXiv:1603.06995,2016.

[25] WANG Z,YAN W,OATES T. Time series classification from scratch with deep neural networks:A strong baseline[C]// 2017 International Joint Conference on Neural Networks(IJCNN). IEEE,2017:1578-1585.

[26] ZHANG H,WANG Z,LIU D. Robust stability analysis for interval Cohen-Grossberg neural networks with unknown time-varying delays[J]. IEEE Transactions on Neural Networks,2008,19(11):1942-1955.

[27] GHASEDI DIZAJI K,HERANDI A,DENG C,et al. Deep clustering via joint convolutional autoencoder embedding and relative entropy minimization[C]// Proceedings of the IEEE International Conference on Computer Vision. 2017:5736-5745.

[28] GUO X,GAO L,LIU X,et al. Improved deep embedded clustering with local structure preservation[C]// IJCAI. 2017:1753-1759.

[29] ZAGORUYKO S,KOMODAKIS N. Learning to compare image patches via convolutional neural networks[C]// Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2015:4353-4361.

[30] CHEN X,HE K. Exploring simple siamese representation learning[C]// Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2021:15750-15758.

[31] YOU Y, CHEN T, SUI Y, et al. Graph contrastive learning with augmentations[J]. Advances in Neural Information Processing Systems, 2020, 33: 5812-5823.
[32] HADSELL R, CHOPRA S, LECUN Y. Dimensionality reduction by learning an invariant mapping[C]// 2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'06). IEEE, 2006: 1735-1742.
[33] OORD A, LI Y, VINYALS O. Representation learning with contrastive predictive coding[J]. arXiv: 1807.03748, 2018.
[34] CHENG P, HAO W, DAI S, et al. Club: A contrastive log-ratio upper bound of mutual information[C]// International Conference on Machine Learning. PMLR, 2020: 1779-1788.
[35] ELDELE E, RAGAB M, CHEN Z, et al. Time-series representation learning via temporal and contextual contrasting[J]. International Joint Conference on Artificial Intelligence. IJCAI, 2021, 2352-2359.
[36] CHEN T, KORNBLITH S, NOROUZI M, et al. A simple framework for contrastive learning of visual representations [C]// International Conference on Machine Learning. PMLR, 2020: 1597-1607.
[37] PEDREGOSA F, VAROQUAUX G, GRAMFORT A, et al. Scikit-learn: Machine learning in Python[J]. The Journal of Machine Learning Research, 2011, 12: 2825-2830.
[38] XIE J, GIRSHICK R, FARHADI A. Unsupervised deep embedding for clustering analysis[C]// International Conference on Machine Learning. PMLR, 2016: 478-487.
[39] BO D, WANG X, SHI C, et al. Structural deep clustering network[C]// Proceedings of the Web Conference 2020. 2020: 1400-1410.
[40] VAN DER MAATEN L, HINTON G. Visualizing data using t-SNE[J]. Journal of Machine Learning Research, 2008, 9(86): 2579-2605.

YANG Bo, born in 1998. postgraduate. His main research interest is time series analysis.

PENG Furong, born in 1987, Ph.D, associate professor, master supervisor. His main research interests include data mining and recommendation systems.

(责任编辑:何杨)

2023 年度 CCF 城市会员活动中心评选结果公布

CCF 城市会员活动中心是联络各地会员并向其提供专业服务的重要节点。CCF 从 2012 年开始创建活动中心,旨在为会员提供"家门口"服务,加强本地会员的交流,分享 CCF 资源。

根据《CCF 会员活动中心条例》,CCF 对 37 个会员活动中心进行了 2023 年度工作评估(成立未满一年的会员活动中心不参与本次评估),通过对会员活动中心活动开展和会员服务、会员发展、对总部活动的参与和支持等三个方面进行客观评分,评审组进行主观评分,综合排名后,评选结果如下:

2023 年度 CCF 优秀会员活动中心:杭州、西安、合肥、哈尔滨、长春、郑州

2023 年度 CCF 优秀会员活动中心 特色活动专项奖:广州、深圳

2023 年度 CCF 优秀会员活动中心 会员发展优秀奖:长沙、上海

2023 年度 CCF 优秀会员活动中心 总部特别贡献奖:沈阳、苏州、太原

2023 年度 CCF 优秀会员活动中心 进步专项奖:兰州

2023 年度 CCF 优秀会员活动中心 会员发展特别贡献奖:西安、兰州、长春、保定、南宁、武汉、郑州、杭州、珠海、宁波、福州、贵阳、济南、石家庄、青岛

2023 年度完成会员发展目标的会员活动中心:西安、兰州、长春、保定、南宁、武汉、郑州、杭州、珠海、宁波、福州、贵阳、济南、石家庄、青岛、上海、长沙、呼伦贝尔、哈尔滨、广州、重庆、南京、南昌、昆明、东莞、乌鲁木齐、常州、太原、西宁、无锡

据 CCF 微信公众号