计算机科学
COMPUTER SCIENCE

基于时间卷积网络的云平台负载预测方法

李英豪, 郭昊龚, 刘盼盼, 相毅浩, 刘成明

引用本文

李英豪, 郭昊龚, 刘盼盼, 相毅浩, 刘成明基于时间卷积网络的云平台负载预测方法[J]. 计算机科学, 2023, 50(7): 254-260.

LI Yinghao, GUO Haogong, LIU Panpan, XIANG Yihao, LIU Chengming. Cloud Platform Load Prediction Method Based on Temporal Convolutional Network [J]. Computer Science, 2023, 50(7): 254-260.

计算机科学 COMPUTER SCIENCE
http://www.jsjkx.com
DOI:10.11896/jsjkx.220500036

基于时间卷积网络的云平台负载预测方法

李英豪　郭昊龚　刘盼盼　相毅浩　刘成明
郑州大学网络空间安全学院　郑州 450000
(yinghaoli@zzu.edu.cn)

摘　要　针对云平台资源负载数据高度非平稳以及存在着随机噪声导致预测准确度较低等问题,结合信号分解和深度学习等技术,提出了一种云平台资源负载预测方法。首先利用经验模态分解(Empirical Mode Decomposition,EMD)方法对原始数据进行分解,得到多个IMF分量;然后构建出基于时间卷积网络(Temporal Convolutional Network,TCN)的预测模型,分别对IMF分量进行预测;最后将预测结果进行合并以得到最终的预测值。将所提方法与传统的预测方法及深度学习预测方法进行比较,并在阿里巴巴开源的数据中心资源监控日志数据集上进行了对比实验。实验结果表明,所提方法的预测误差分别比ARIMA,Bi-LSTM,GRU,TCN降低了36.75%,23.5%,24.44%,24.53%,预测结果具有最优的准确度。

关键词:云计算;负载预测;时间卷积网络;经验模态分解
中图法分类号　TP393

Cloud Platform Load Prediction Method Based on Temporal Convolutional Network

LI Yinghao,GUO Haogong,LIU Panpan,XIANG Yihao and LIU Chengming
School of Cyber Science and Engineering,Zhengzhou University,Zhengzhou 450000,China

Abstract　Aiming at the problems of highly non-stationary cloud platform resource load data and low prediction accuracy due to random noise,a cloud platform resource load prediction method is proposed by combining signal decomposition and deep learning technologies. Firstly,the original data is decomposed using empirical mode decomposition(EMD) method to obtain multiple IMF components;then a prediction model based on temporal convolutional network(TCN) is constructed to predict the IMF components separately;finally,the prediction results are combined to obtain the final prediction value. The proposed method is compared with traditional prediction methods and deep learning prediction methods,and a comparative experiment is carried out on Alibaba's open source data center resource monitoring log data set. Experimental data results show that the prediction errors of the proposed method reduces by 36.75%,23.5%,24.44%,and 24.53% compared with ARIMA,Bi-LSTM,GRU and TCN,respectively,and the prediction results have the best accuracy.

Keywords　Cloud computing,Load prediction,Temporal convolutional network,Empirical mode decomposition

1　引言

2006年,Google[1]在搜索引擎大会上首次提出了云计算这一概念,这是继分布式处理、并行处理和网格计算的又一计算模式。云计算的出现彻底改变了传统计算机资源的使用方式,它将所有的软硬件资源都放在云端,并且提供了一种pay-as-you-go(按需付费)[2]的服务模式。然而,云计算平台中的资源利用率较低,根据相关统计[3],云计算数据中心的资源利用率仅维持在10%～50%。这种资源未被合理利用的服务器通常被称为"僵尸"服务器[4],这些"僵尸"服务器的资源利用率一直处于较低的水平,大量的资源被严重浪费。同时,浪费的资源消耗着巨大的能源并产生了更多的污染物。

针对资源利用率较低的问题,通常通过弹性伸缩机制来解决。弹性伸缩一般分为被动和主动两种方式[5]。被动的方式是当系统的资源利用率达到定义的阈值时触发弹性伸缩操作,然而这种方式的时效性较差。主动的方式是建立某种预测模型,根据历史的负载数据提前预测出未来时刻所需的资源量,并进行弹性伸缩,最大化保证资源的合理利用。这种主动预测的方式有效地弥补了被动方式的不足,但是预测的准确率却面临着巨大挑战。一般来说,云平台中的负载的变化会受到许多随机因素的影响,如每年618、双11电商节可能会造成资源使用量的急剧增加。此外,Google于2011年公开的云平台资源数据表示[6],CPU、内存等资源是高度随机变化的,这均使得负载的预测工作较难进行。

目前,关于云平台负载预测的研究方法非常多,可以分成传统的时间序列方法和基于学习的方法。传统的时间序列

到稿日期:2022-05-05　返修日期:2022-09-24
基金项目:国家重点研发计划(2020YFB1712401)
This work was supported by the National Key R&D Program of China(2020YFB1712401).
通信作者:刘成明(cmliu@zzu.edu.cn)

方法[7]一般是基于参数模型，在确定了所使用的预测模型后求解其参数，待参数确定后使用模型进行预测，主要包括自回归(Autoregressive model，AR)[8]、移动平均(Moving Average model，MA)[9]、自回归移动平均(Autoregressive Moving Average model，ARMA)[10]、差分自回归移动平均(Autoregressive Integrated Moving Average model，ARIMA)[11]等方法。

基于学习的方法主要是通过为模型提供带有标签的数据，使用这些数据不断调整模型参数，使模型产生推理能力，从而使得未知的数据能够映射出正确的结果[12]。许多研究人员提出基于机器学习的预测方法。例如，Hsieh 等[13]采用灰色马尔可夫模型来进行资源的预测，实验结果表明在保证云计算的服务质量的前提下能减少动态虚拟机的迁移次数和能耗。Zhong 等[14]提出了一种结合小波变换和支持向量机的方法，并使用粒子群优化算法进行参数优化，该模型显著地提高了预测准确度。Peng 等[15]提出了一种鲸鱼优化算法结合极限学习机的预测模型，该模型具有很强的非线性映射能力。Tofighy 等[16]使用贝叶斯信息准则进行 CPU 负载的预测，并且使用平滑过滤器来减小数据中异常值对预测的影响，有效地提升了预测精度。Rahmanian 等[17]提出了一种基于学习自动机的模型，预测结果取得了不错的表现。

近年来，随着深度学习的发展，大量的研究人员将深度学习模型应用于预测领域[18-21]。例如，Song 等[22]使用了长短期记忆网络进行主机负载预测，并取得了不错的效果。Nguyen 等[23]在长短期记忆网络的基础上添加编码器-解码器，以提高模型的记忆能力，并且与单层 LSTM 和多层 LSTM 模型进行实验对比，模型的预测精度更好。Karm 等[24]提出了一种混合循环神经网络预测模型，该模型结合了长短期记忆和门控循环单元，能够很好地增强非线性的数据分析能力，实验的预测精度高于传统的单一模型。

无论是传统的时间预测方法还是基于学习的方法，都没有考虑到预测数据的非平稳性，对波动较大的数据直接进行预测则会导致预测准确度偏低。此外，大多数预测方法都是通过单一模型进行预测，这样的好处是训练时间较短，但预测精度仍不够好。因此，针对上述问题，本文提出了一种基于时间卷积网络的云平台负载预测方法。通过信号处理中的分解方法将云平台中原始的非平稳负载数据分解为多个平稳分量，然后分别对其进行预测，相比传统的直接在原始非平稳数据上进行训练，预测准确度更高；将时间卷积网络模型和多头注意力机制相结合，提高模型对历史数据在时间上的依赖关系的学习能力；并使用软阈值函数作为模型中的激活函数，既能有效解决神经元坏死问题，又能提高模型的抗噪能力。实验结果表明，与传统的预测方法相比，本文方法能够在云平台数据非平稳、存在大量随机噪声的情况下有效提高预测准确率。

2 E-SMT 云平台负载预测模型

为了对云平台中非平稳、高动态变化的数据进行准确预测，本文提出了一种经验模态分解(EMD)[25]结合软阈值函数(Soft Thresholding)[26]以及多头注意力机制(Multi-head-attention)[27]的时间卷积网络模型，简称 E-SMT 模型，模型的整体结构如图 1 所示。

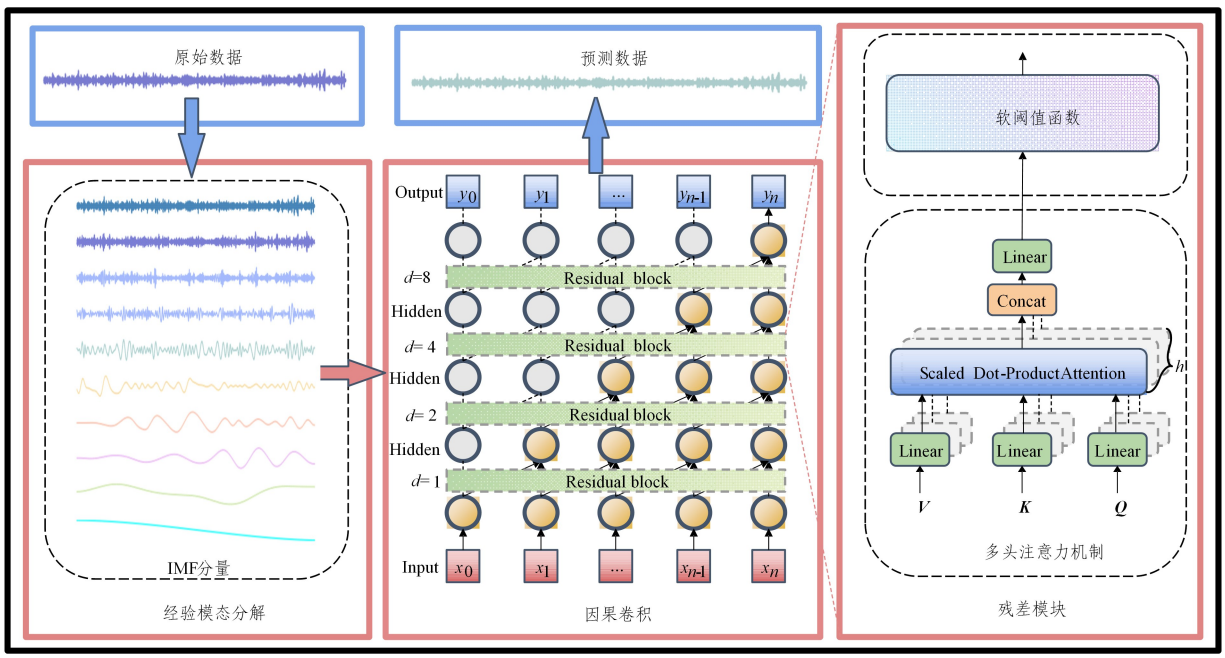


图 1 E-SMT 模型

Fig. 1 E-SMT model

首先，本文引入了经验模态分解方法，它可以将原始数据分解为多个平稳的分量。然后，将分解得到的数据输入因果卷积(Causal Convolution)网络中。在因果卷积模块中，使用软阈值函数进一步提高模型对含有噪声数据的预测能力，同时引入多头注意力机制来增强模型的非线性拟合能力。以下将详细介绍本文方法的主要工作。

2.1 经验模态分解

云平台数据中心的负载数据并不是平稳变化的，往往一些突发事件可能会导致资源使用量的波动。例如，数据中心的设备故障可能会导致负载数据使用量的减少，网络延时也

可能使资源使用量降低等。在这些情况下，时间序列则表现为骤变的极值点，这会使得预测难度剧增。另一个问题是云平台的负载数据在采集的过程中也可能会引入大量噪声，这也可能会使原始数据的准确度急剧降低，直接预测这种含有大量噪声的数据很难得到精确的结果。针对这种问题，本节引入了信号处理中的经验模态分解方法，将原始序列分解为多个特征序列和残差序列，以充分提取出数据的特征，从而提高预测的准确度。

经验模态分解的关键是将复杂的信号分解成多个本征模函数(Intrinsic Mode Function，IMF)。它存在着两个约束条件，一个是整个数据中，极值点和零点的个数必须相等或者最多相差一个；另一个是任意时刻的上包络线和下包络线的平均值必须为零。

EMD 的分解过程如下：

(1)识别出原始数据 $x(t)$ 的所有极值点，通过插值法并利用极大值点和极小值点分别得到上包络线 $u(t)$、下包络线 $l(t)$。

(2)求出上、下包络线的均值 $m(t)=\frac{u(t)+l(t)}{2}$，并根据 $x(t)-m(t)$ 得到中间信号 $h(t)$。

(3)检查 $h(t)$ 是否满足 IMF 的条件。如果满足，则将该 IMF 分量记为 $c_1(t)$；否则，将 $h(t)$ 作为新的信号，并不断重复步骤(1)和步骤(2)，直到新的中间信号满足 IMF 的条件，记为 $c_1(t)$。

(4)根据 $x(t)-c_1(t)$ 得到剩余分量(Residual Component)，记为 $r_1(t)$。将剩余分量作为新的原始信号，并不断重复步骤(1)－步骤(3)得到 $c_n(t)$，直到 $r_n(t)$ 无法分解时结束。

最终，原始数据经过分解可表示为式(1)：

$$x(t)=\sum_{i=1}^{n}c_i(t)-r_n(t) \tag{1}$$

其中，$c_1(t)$ 到 $c_n(t)$ 分别是频率由高到低的 IMF 分量。

本文将云平台负载数据分解为多个时间序列和残差，分解后的子序列具有较好的平稳性和规律性，这比直接预测原始序列的精确度更高，结果表现更好。

2.2 因果卷积

时序建模大多都是采用循环神经网络(RNN)及其变种方法进行的，RNN 一般很难将信息长期保存，而且还存在着梯度消失和梯度爆炸问题，因此预测的效果并不是很好。LSTM 和 GRU 的出现很好地解决了 RNN 的长期依赖和梯度消失问题，它们能够通过门控机制对信息有选择地记忆或是遗忘，这对于长序列建模具有不错的效果。它们的缺点是并行性较差、训练内存消耗大，而时间卷积网络(TCN)正好能够弥补上述方法的缺陷，它具有很多优点，如并行性很好，不需要顺序地进行处理，给定输入即可并行运算；具有稳定的梯度，不易出现梯度消失等问题；占用内存低，传统 RNN 模型需要将每步计算的中间结果保存起来，而 TCN 的卷积核是共享的，意味着内存的占用更低。

TCN 包含了 3 个基本的模块，分别是因果卷积、膨胀卷积和残差连接。本节基于时间卷积网络，利用因果卷积并引入多头注意力机制及软阈值化函数来提高模型的预测能力。

因果卷积将普通的卷积操作限定在一个方向，时刻 t 的输出只能来自 $t-1$ 时刻及以前的输入。从整体来看，网络的输出只取决于历史数据，而不会泄露未来的特征。它的结构如图 2 所示。

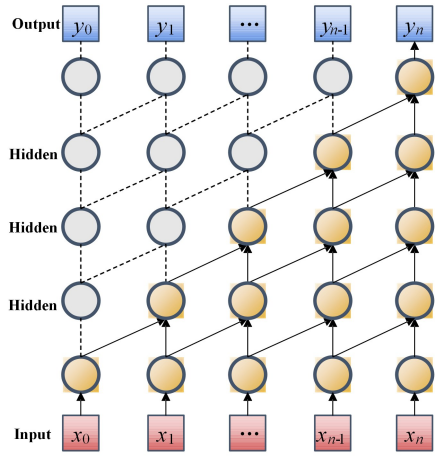


图 2 因果卷积

Fig. 2 Causal convolution

在 input 中，每个位置对应一个历史数据。将历史数据输入网络中，与 $2*1$ 的卷积核做卷积操作，在每层从左向右依次移动，最终传递到顶层即可得到因果卷积的结果。因果卷积存在着一些缺陷，为了能够捕捉到更长的历史信息，网络的层数也必须增加，这很可能会出现梯度消失和梯度爆炸等问题，而这种问题可在因果卷积的基础上引入残差连接来解决。

2.3 多头注意力机制

深层的网络具有强大的特征提取能力和非线性拟合能力，但是往往很容易出现梯度消失和梯度爆炸问题。残差连接被证明非常适合于非常深的网络，它通过跨层连接的方式，将下层的信息恒等映射到上层，这样做的好处是随着网络深度的增加，特征信息不会减弱，这能够很好地解决梯度消失和梯度爆炸问题。本节基于残差网络并使用多头注意力机制来增强模型的非线性拟合能力，它对于学习时间序列的长距离依赖关系十分有效。多头注意力机制如图 3 所示。

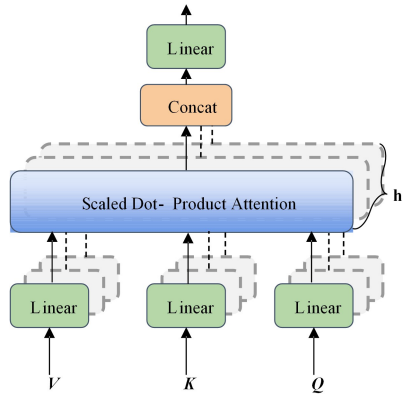


图 3 多头注意力机制

Fig. 3 Multi-head-attention

多头注意力机制基于缩放点积注意力(Scaled Dot-Product Attention)实现，缩放点积注意力如式(2)所示。首先输入 d_k 维度的查询 $\boldsymbol{Q}$、键 $\boldsymbol{K}$，以及 d_V 维度的值 $\boldsymbol{V}$，然后计算出

所有的$\boldsymbol{Q}$和$\boldsymbol{K}$的点积，再除以$\sqrt{d_k}$并输入到Softmax函数中，最后得到值的权重。

$$Attention(\boldsymbol{Q},\boldsymbol{K},\boldsymbol{V})=\mathrm{softmax}\left(\frac{\boldsymbol{Q}\boldsymbol{K}^{\mathrm{T}}}{\sqrt{d_k}}\right)\boldsymbol{V} \tag{2}$$

其中，$\boldsymbol{Q},\boldsymbol{K},\boldsymbol{V}$分别代表查询、键、值3个矩阵；$\frac{1}{\sqrt{d_k}}$为缩放因子，保证内积不会过大。

在缩放点积注意力的基础上，多头注意力机制首先将$\boldsymbol{Q},\boldsymbol{K},\boldsymbol{V}$进行线性变换，然后引入缩放点积注意力机制，通过式(3)得到h个不同的表示，最后将它们连接并做线性变换得到最终结果。

$$MultiHead(\boldsymbol{Q},\boldsymbol{K},\boldsymbol{V})=Concat(head_1,\cdots,head_h)W^O \tag{3}$$

其中，$head_i$如式(4)所示：

$$head_i=Attention(\boldsymbol{Q}\boldsymbol{W}_i^Q,\boldsymbol{K}\boldsymbol{W}_i^K,\boldsymbol{V}\boldsymbol{W}_i^V) \tag{4}$$

2.4 软阈值函数

云平台数据中心的负载数据在采集的过程中往往会引入大量噪声，直接预测这种含有大量噪声的数据很难得到精确的结果。因此，本节引入软阈值函数以去除模型训练过程中的噪声数据。软阈值函数可表示为：

$$y=\begin{cases}x-\tau, & x>\tau\\ 0, & -\tau\leqslant x\leqslant\tau\\ x+\tau, & x<-\tau\end{cases} \tag{5}$$

其中，x是输入的特征，τ是阈值，y是输出的特征，软阈值函数将接近零的特征设置为零，这样能够有效地过滤掉无用的噪声数据，大大提高了模型的预测精度。

软阈值函数如图4所示。

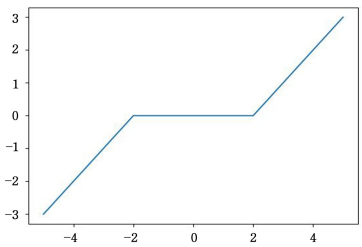


图4 软阈值函数

Fig. 4 Soft threshold function

除了用于去除数据在模型训练过程中的噪声之外，软阈值函数还具有激活函数的功能。该函数的导数如式(6)所示，它的值只有0和1。这与ReLU激活函数性质相似，它不仅保留了ReLU激活函数的一些优点，如不存在梯度饱和问题、计算速度快等，而且也能够避免ReLU激活函数的缺点，如当输入为负时，会产生神经元坏死问题(Dead ReLU Problem)。

$$\frac{\partial y}{\partial x}=\begin{cases}1., & x>\tau\\ 0, & -\tau\leqslant x\leqslant\tau\\ 1, & x<-\tau\end{cases} \tag{6}$$

3 实验

3.1 实验数据集

为了验证E-SMT模型对云平台负载的预测效果，本文在Alibaba cluster trace[28]数据集上进行相关实验，其中包含了8天内大约4000台机器运行所产生的负载资源。该数据内包含以下信息：时间戳(timestamp)、机器ID(machineID)、CPU利用率(util:CPU)、内存利用率(util:memory)、磁盘利用率(util:disk)等，所有的资源利用率都是以百分比表示。本实验随机选取了一台主机的负载数据，并选择CPU利用率作为预测目标，从图5中可以看到，该数据集在8天内的负载利用率在20%～80%之间变化，整体具有一定的周期性，但局部变化更加频繁。

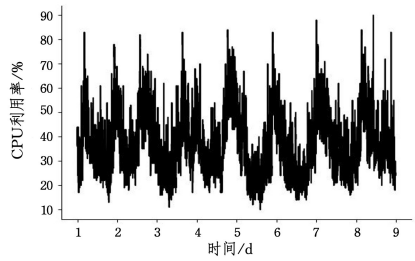


图5 Alibaba cluster trace数据集

Fig. 5 Alibaba cluster trace dataset

本实验使用滑动窗口的方式构建数据集，即通过滑动的方式不断地对固定窗口大小的数据进行切分，分别构成模型的输入数据和真实数据。为了加快神经网络模型的学习速度，本实验对构建完成后的数据进行归一化处理，如式(7)所示：

$$x'=\frac{x-\mathrm{mean}(x)}{\max(x)-\min(x)} \tag{7}$$

其中，x'为归一化后的值，x为原始数据，$\mathrm{mean}(x)$，$\min(x)$和$\max(x)$分别是原始数据的平均值、最小值和最大值。

3.2 对比方法

为了验证本文方法的有效性，选择使用差分自回归移动平均ARIMA[11]、双向长短期记忆网络Bi-LSTM[19]、门控循环单元GRU[20]，以及时间卷积网络模型TCN[21]进行对比。分别使用以上方法进行模型训练，并对测试集进行测试，最后进行结果比较。

3.3 评价指标

为了客观地评价模型的预测准确度，使用平均绝对误差(Mean Absolute Error，MAE)、均方根误差(Root Mean Square Error，RMSE)以及残差平方和(R-Squared)作为实验预测结果的评价指标。

平均绝对误差(MAE)为：

$$MAE=\frac{1}{m}\sum_{i=1}^{m}|(y_i-\hat{y}_i)| \tag{8}$$

均方根误差(RMSE)为：

$$RMSE=\sqrt{\frac{1}{m}\sum_{i=1}^{m}(y_i-\hat{y}_i)^2} \tag{9}$$

残差平方和(R-Squared)为：

$$R^2=1-\frac{\sum_{i=1}^{m}(\hat{y}_i-y_i)^2}{\sum_{i=1}^{m}(\hat{y}_i-y_i)^2} \tag{10}$$

其中，$\hat{y}$是真实的负载数据，y是预测的数据，m是负载数据的数量。对于MAE和RMASE，评价指标的值越低，说明预测的结果越准确，方法就越好。而对于R-Squared，它的上限为1，该评价指标的值越接近1，说明模型的预测准确度越高。

3.4 实验设置

3.4.1 实验环境

本实验的服务器运行环境如表 1 所列。

表 1 运行环境配置

Table 1 Operating environment configuration

实验环境	配置信息
操作系统	Ubuntu 20.04.2 LTS
CPU	Intel(R) Xeon(R) Silver 4210
内存/GB	64
GPU	NVIDIA Tesla T4
编程语言	Python3.6
深度学习框架	PyTorch 1.4.1

3.4.2 参数设置

本实验的参数设置如表 2 所列。

表 2 模型参数配置

Table 2 Model hyperparameter configuration

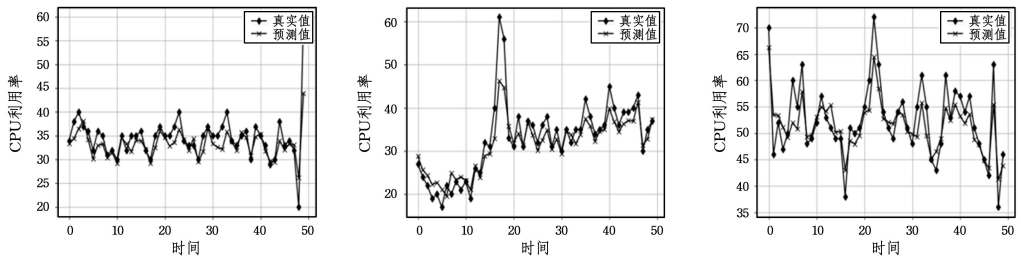
参数名称	参数含义	值
ω	历史窗口长度	80
t	预测窗口长度	1
batchsize	批处理大小	64
epochs	迭代次数	200
ε	学习率	0.001
criterion	损失函数	MSELoss
optimizer	优化器	Adam

本实验所设置的预测长度为单步预测，即通过历史窗口对未来的下一个时间点进行预测。将所有数据集分为训练集(80%)、验证集(10%)和测试集(10%)。训练集用于对整个预测模型进行训练，验证集用于参数选择，测试集用于性能评价。训练时选取 MSELoss 作为损失函数，选择 Adam 作为优化器。

4 结果分析

4.1 实验结果

为了验证本文模型在不同类型数据下的预测效果，分别选择变化有规律的、有突变值的以及波动程度较大的数据进行实验，模型的预测结果如图 6 所示。从图中能够看出，本文的 E-SMT 模型的预测结果和真实数据值的趋势基本吻合。其中，图 6(a)中的数据变化具有一定的规律性且较平稳，整体的预测效果较好；图 6(b)中的数据整体来看仍具有一定的规律性，其中有一处明显的突变值，该处的值变化较大，模型仍然能够预测出变化的趋势，但预测值和真实值也存在着一定的差别；图 6(c)中的数据波动较大，且存在着更多的突变值，但模型在这些突变点处仍然能够预测出变化的趋势。

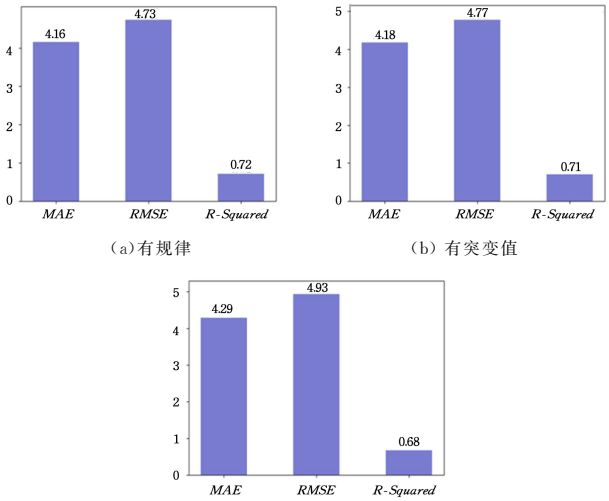


(a)有规律 (b) 有突变值 (c)波动较大

图 6 不同类型数据负载预测结果

Fig. 6 Prediction results of different types of data loads

图 7 给出了不同类型数据的预测评价指标结果。



(a)有规律 (b) 有突变值

(c)波动较大

图 7 评价指标结果

Fig. 7 Evaluation index results

其中，图 7(a)和图 7(b)分别是变化比较有规律的和局部有突变值的负载数据预测结果，其评价指标值差别不大，说明模型对于具有少量局部突变值的数据仍能准确预测；图 7(c)为波动较大、规律不明显且突变值更多的负载数据预测评价指标结果，其 MAE 值和 RMSE 值相比前两种数据有一定的上升，R-Squared 值则有些下降，但变化不大，这说明模型对于波动较大的数据仍然能够保证预测的准确度。对于波动较大的数据，整体的预测准确度有小幅度的下降。分析其原因，主要是在局部突变值处预测的结果和真实值之间出现了较小的偏差，而大量偏差的累积则会导致整体预测精度变差，但本文在后续实验中与传统的预测模型相比，仍具有较好的预测效果。

4.2 消融实验

为了验证本文方法中经验模态分解模块、多头注意力机制模块以及软阈值函数模块对模型整体预测准确率的提升效果，以及模型的抗噪效果，本节分别计算出各个模块的平均绝对误差、均方根误差以及残差平方和的值，结果如表 3 所列。

本文的 E-SMT 模型在时间卷积网络模型的基础上分别引入了经验模态分解、软阈值函数以及多头注意力机制。在最终的结果中，本文模型的预测精度为最优，这表明 E-SMT 模型对于 TCN 模型的改进有一定的效果。在 TCN 的基础

上加入软阈值函数之后，预测的误差值有一定的减小，这说明软阈值函数具有一定的抗噪性。此外，由于软阈值函数能够优化 ReLU 激活函数的神经元坏死问题，因此也进一步提高了预测精度；同时在 TCN 的基础上加入 EMD 经验模态分解可将原始的非平稳数据分解为多个平稳数据，而对于平稳数据的预测易取得优良的效果；多头注意力机制能够使模型充分学习数据在历史时间上的依赖关系，在一定程度上提高了模型的预测能力，因此本文模型在预测精度上有一定的提升。

表 3 各模块预测结果对比

Table 3 Module prediction results comparison

Module	*MAE*	*RMSE*	*R-Squared*
TCN	4.8785	6.5356	0.5825
STCN	4.7967	6.2417	0.6011
MTCN	4.8343	6.3130	0.5912
E-TCN	4.3212	5.2133	0.6631
E-SMT	4.2907	4.9325	0.6832

4.3 对比实验

为了验证模型的有效性，本文选择与 ARIMA[11]，Bi-LSTM[19]，GRU[20] 和 TCN[21] 进行对比，各模型对 CPU 负载的预测情况如图 8 所示。

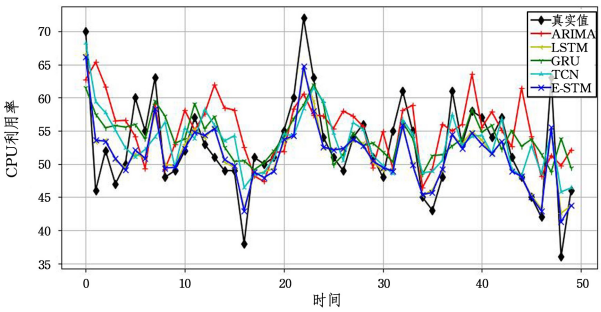


图 8 各模型负载预测结果

Fig. 8 Load prediction results of each model

表 4 列出了不同模型进行负载预测实验的 MAE，RMSE，R-Squared 值以及模型训练所消耗的时间。

表 4 模型评价指标表

Table 4 Model evaluation index

Module	*MAE*	*RMSE*	*R-Squared*	Time/s
ARIMA	6.1513	7.7986	0.4394	67
Bi-LSTM	4.7069	6.4478	0.6306	65
GRU	4.7281	6.5282	0.6135	**61**
TCN	4.8785	6.5356	0.5825	75
E-SMT	**4.2907**	**4.9325**	**0.6832**	83

表 4 中，ARIMA 模型的预测精度相对来说最差，分析其原因，主要是 ARIMA 要求数据是平稳的并且其只能捕捉线性关系，而本文实验数据为非平稳数据，因此预测效果较差。而 Bi-LSTM，GRU 以及 TCN 模型的预测精度相差不大，Bi-LSTM 略优于 GRU 和 TCN 模型，虽然 GRU 模型是在 LSTM 模型的基础上改进而来的，但由于 GRU 减少了一些门控单元且参数更少，因此 GRU 模型的提取特征能力略差于 LSTM 模型，预测精度略低，但 GRU 模型的训练时间相对最短。E-SMT 模型的平均绝对误差和均方根误差值均小于传统预测模型，而残差平方和的值大于传统模型，这说明本文的预测模型优于传统的预测模型。但由于本文模型的参数量较大，因此训练时间相对较长。

结束语 本文提出了一种基于时间卷积网络的云平台负载预测模型——E-SMT，旨在解决云平台资源数据高度非平稳以及存在着噪声导致预测准确度低的问题。传统的预测模型大多都是基于单一模型，并且没有充分考虑噪声数据对预测结果的影响，而本文使用经验模态分解方法将原始非平稳数据分解为多个平稳数据以提高预测准确度，同时基于时间卷积网络并利用多头注意力机制来充分学习历史数据在时间上的依赖关系，最后通过软阈值化函数来减小模型训练过程中噪声数据对结果的影响。在 Alibaba cluster trace 数据集上进行实验并与传统的 ARIMA，Bi-LSTM，GRU 以及 TCN 模型进行对比，本文提出的负载预测模型具有较好的预测精度。但本文模型参数量较大，导致训练速度较慢，下一步的工作考虑采用深度学习模型压缩与加速方法，如知识蒸馏、紧凑网络、参数剪枝等方法来减少模型的参数量，以加快模型的训练速度。

参 考 文 献

[1] JADEJA Y，MODI K. Cloud computing-concepts，architecture and challenges[C]// 2012 International Conference on Computing，Electronics and Electrical Technologies(ICCEET). Nagercoil：IEEE，2012：877-880.

[2] MIYACHI C. What is "Cloud"? It is time to update the NIST definition? [J]. IEEE Cloud Computing，2018，5(3)：6-11.

[3] ZHANG C，WANG Y，LV Y，et al. An energy and sla-aware resource management strategy in cloud data centers[J]. Scientific Programming，2019，2019：1-16.

[4] PANNEERSELVAM J，LIU L，HARDY J，et al. Analysis，Modelling and Characterisation of Zombie Servers in Large-Scale Cloud Datacentres[J]. IEEE Access，2017，5：15040-15054.

[5] MORENO-VOZMEDIANO R，MONTERO R S，HUEDO E，et al. Efficient resource provisioning for elastic Cloud services based on machine learning techniques[J]. Journal of Cloud Computing，2019，8(1)：1-18.

[6] REISS C，WILKES J，HELLERSTEIN J L. Google cluster-usage traces：format+ schema [J]. Google Inc.，White Paper，2011，1：1-14.

[7] YANG H M，PAN Z S，BAI W. Review of Time Series Prediction Methods[J]. Computer Science，2019，46(1)：21-28.

[8] ZHI G W，GAO M. Parameter estimation of random coefficient autoregressive model with missing data [J]. Statistics & Decision，2022，38(1)：16-20.

[9] XIE Y，JIN M，ZOU Z，et al. Real-time prediction of docker container resource load based on a hybrid model of ARIMA and triple exponential smoothing[J]. IEEE Transactions on Cloud Computing，2020，10(2)：1386-1401.

[10] SUN Y，SONG X Y，JIN L T，et al. Railway Passenger Flow Forecast Based on Armah stm combined model [J]. Computer Applications and Software，2021，38(12)：262-267，273.

[11] WU F，JING R，ZHANG X P，et al. A combined method of improved grey BP neural network and MEEMD-ARIMA for day-

ahead wave energy forecast[J]. IEEE Transactions on Sustainable Energy, 2021, 12(4): 2404-2412.
[12] OPREA S, MARTINEZ-GONZALEZ P, GARCIA-GARCIA A, et al. A review on deep learning techniques for video prediction [J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2020, 44(6): 2806-2826.
[13] HSIEHSY, LIUCS, BUYYAR, etal. Utilization-prediction-aware virtual machine consolidation approach for energy-efficient cloud data centers[J]. Journal of Parallel and Distributed Computing, 2020, 139: 99-109.
[14] ZHONG W, ZHUANG Y, SUN J, et al. A load prediction model for cloud computing using PSO-based weighted wavelet support vector machine[J]. Applied Intelligence, 2018, 48(11): 4072-4083.
[15] PENG H, WEN W S, TSENG M L, et al. A cloud load forecasting model with nonlinear changes using whale optimization algorithm hybrid strategy[J]. Soft Computing, 2021, 25(15): 10205-10220.
[16] TOFIGHY S, RAHMANIAN A A, GHOBAEI-ARANI M. An ensemble CPU load prediction algorithm using a Bayesian information criterion and smooth filters in a cloud computing environment[J]. Software: Practice and Experience, 2018, 48(12): 2257-2277.
[17] RAHMANIAN A A, GHOBAEI-ARANI M, TOFIGHY S. A learning automata-based ensemble resource usage prediction algorithm for cloud computing environment[J]. Future Generation Computer Systems, 2018, 79: 54-71.
[18] LIU B, WANG M S, LI Y, et al. Deep Learning for Spatio-Temporal Sequence Forecasting: A Survey[J]. Journal of Beijing University of Technology, 2021, 47(8): 925-941.
[19] NI X L, SHI C A, M Y L, et al. Research on fault prediction method of electronic Equipment based on BI-LSTM[J]. Aero Weaponry, 2022, 29(6): 102-110.
[20] CAO G H, ZHAO Z L, XU Y H. Research on Health State Prediction of Lithium Battery Pack Based on GRU[J]. Journal of Jilin University(Information Science Edition), 2022, 40(2): 181-187.
[21] BAI S, KOLTER J Z, KOLTUN V. An empirical evaluation of generic convolutional and recurrent networks for sequence modeling[J]. arXiv: 1803.01271, 2018.
[22] SONG B, YU Y, ZHOU Y, et al. Host load prediction with long short-term memory in cloud computing[J]. The Journal of Supercomputing, 2018, 74(12): 6554-6568.
[23] NGUYEN H M, KALRA G, KIM D. Host load prediction in cloud computing using long short-term memory encoder-decoder [J]. The Journal of Supercomputing, 2019, 75(11): 7592-7605.
[24] KARIM M E, MASWOOD M M S, DAS S, et al. BHyPreC: a novel Bi-LSTM based hybrid recurrent neural network model to predict the CPU workload of cloud virtual machine[J]. IEEE Access, 2021, 9: 131476-131495.
[25] FLANDRIN P, RILLING G, GONCALVES P. Empirical mode decomposition as a filter bank[J]. IEEE Signal Processing Letters, 2004, 11(2): 112-114.
[26] DONOHO D L. De-noising by soft-thresholding [J]. IEEE Transactions on Information Theory, 1995, 41(3): 613-627.
[27] VASWANI A, SHAZEER N, PARMAR N, et al. Attention is all you need[J/OL]. Advances in Neural Information Processing Systems, 2017, 30. https://papers.nips.cc/paper_files/paper/2017/hash/3f5ee243547dee91fbd053c1c4a845aa-Abstract.html.
[28] GUO J, CHANG Z, WANG S, et al. Who limits the resource efficiency of my datacenter: An analysis of alibaba datacenter traces[C]// 2019 IEEE/ACM 27th International Symposium on Quality of Service(IWQoS). Phoenix, AZ, USA: IEEE, 2019: 1-10.

LI Yinghao, born in 1987, Ph.D, lecturer, master supervisor, is a member of China Computer Federation. His main research interests include machine learning and data mining.

LIU Chengming, born in 1979, Ph.D, assistant professor, master supervisor, is a member of China Computer Federation. His main research interests include computer vision and cloud computing.

(责任编辑:喻藜)