摘要

集群运行时跟踪日志（trace）数据因蕴含系统的负载波动、节点配置等重要的时空维度信息，是集群性能分析的最重要的数据集之一。对集群trace的时序数据挖掘研究存在两方面问题：一方面，现有的在计算机系统性能分析领域从时序数据挖掘角度开展的研究依然相对缺乏并且不够深入；另一方面，由于数据规模、数据特征和应用场景等方面存在的巨大差异，其它领域基于深度学习的时序数据挖掘算法并不能直接应用到集群trace 时序数据上来。

针对上述问题，本文研究基于深度学习的集群trace 时序数据挖掘算法，主要工作和创新点包括：

1）在负载转折点预测方面，针对现有方法中特征表达能力较弱、且基于单一时间窗口的特征构造方法无法描述时间维度的变化模式这两个问题，提出了一种特征增强的多任务LSTM 负载转折点预测算法。

2）在任务失败预测方面，针对监测指标序列内部和各检测指标序列之间存在的两种不同的特征难以同时提取的问题，提出了一种特基于通道注意力卷积模型的任务失败预测算法。

3）在I/O 负载预测方面，针对传统机器学习方法难以捕捉强烈波动的I/O 负载时序特征这个问题，提出了一种特基于长短期记忆网络的存储系统I/O 负载短期时序预测算法。

国内外相关研究及技术

2.1集群trace数据

trace 数据按照采集的场景可以分为：存储系统I/O trace，云服务作业trace，高性能计算trace 等。其中云服务作业场景下的trace 研究较多，谷歌在2012 年发布的trace 数据是一个典型的代表。它主要由Job events table、Task events table、Task constraints table、machine event table、Machine attributes table、Resource usage 6 张表组成。

对于提交到集群中的作业，首先会在Job events table 中记录一个作业提交事件，包含作业的提交时间、提交用户、作业的调度类别。然后作业被拆分为一个或者多个任务，当任务被调度时会在Task events table 中记录对应的调度动作，Resource usage table 中则记录了每个任务在间隔为5 分钟的测量周期内的各种资源消耗数据，包括CPU、内存使用率等。

2.2 负载转折点预测问题相关研究及技术

2.2.1相关研究

目前的云环境下负载研究主要集中在点值预测，转折点预测的研究相对较少。

2018年Xia 等人[10] 针对云容量规划（Cloud capacity planning）问题进行研究，提出用户对虚拟机请求总量的剧烈波动，往往会造成云服务供应商和和用户之间的不匹配现象。该研究认为对虚拟机请求量进行点值预测无法应对这种突变点，所以提出了PLR-WSVM 模型，专门对突变点也即转折点进行预测。

该模型首先采用自顶向下的分段线性分割算法PLR[11] 对虚拟机请求量的时间序列进行转折点标记，然后通过WSVM[12] 模型来预测下一刻的负载点是否是突变点。模型首先设置了一个固定大小的观测窗口，包含了历史的请求量序列，然后使用两类特征来描述虚拟机请求数量的变化：第一类是虚拟机请求的元信息（meta information），第二类特征是观测窗口中的统计特征。这些特征被用来预测下一个点是否是转折点。

上述研究存在几处不足：首先，作者所使用的特征大都是时间窗口内基础的统计特征，如均值，方差等，这些基础特征在表现负载波动性方面能力较弱；其次，基于单个窗口中统计特征的建模方法可能无法捕捉请求序列在时间维度的变化规律，下一时刻的取值可能受到过去取值的影响；最后，作者采用的数据集规模太小。

2.2.2相关技术

在文献中[12] 转折点是局部趋势的变化点，常常需要通过时间序列的线性分段算法[11]（PLR）进行标记，PLR 分割算法一般根据边界切分略分为三类：滑动窗口、自顶向下和自底向上。

滑动窗口分割算法的基本思路是首先设置一个阈值，然后开始不断地将下一个点纳入当前窗口内，每次加入一个点值之后就对窗口内数据拟合一条直线并计算拟合误差。

自顶向下的算法考虑了时间序列的每一个可能的分割位置。首先也需要设定一个阈值，然后遍历整条时间序列中除两端点外的其他位置，对于每一个位置，分别以该位置和两端点作为新片段的端点，拟合一条直线并分别计算拟合误差。

自底向上分的算法同样需要设定一个阈值，它每次计算合并相邻的两个片段之后新片段的拟合误差，遍历所有相邻片段之后找出误差最小的两个相邻片段，然后将其合并。直到该最小误差大于设定的阈值为止。

2.3任务失败预测问题相关研究及技术

2.3.1相关研究

目前集群任务失败预测方法主要采用时间序列分类技术，通过任务运行过程中采集的监测信息，构建时序模型从而提取时序特征以进行预测。

2009 年Fadishei[17] 等人研究了网格环境下CPU 利用率、内存使用率等指标和任务是否会失败这两者之间的相关性。Pan 等人[18] 使用黑盒错误诊断的方法来诊断MapReduce 系统的错误，发现失败节点和正常节点之间存在着不同的行为模式。

2014 年Chen 等人[13] 在谷歌集群数据集[3] 上进行研究，使用每个任务执行时消耗的资源情况来描述任务的特征，使用深度学习模型中的循环神经网络来进行特征提取，以预测一个任务最后是否会失败。

2016 年刘春红[19] 等人主要研究云集群下作业的失败预测问题。该研究将谷歌集群数据中作业提交时的属性看做静态特征，将作业中各个任务的监测指标作为动态特征，并采用SVM 作为分类器以预测失败作业。由于谷歌trace 中作业的静态特征有较多的缺失，该研究的试验规模较小；且该研究以作业作为研究对象粒度较粗。

2017 年Islam[14] 等人延续了Chen 等人[13] 的工作，他们扩展了任务的特征，加入了一些静态信息，并采用深度学习模型中更为先进的长短期记忆网络来提取时间序列蕴含的特征，最后通过全局平均池化层，从长短期记忆网络所有时间步输出的隐层表示中提取执行失败任务的全局时序特征，以进行预测。Islam 等人扩展的任务静态特征有些是不合理的。

整体上Chen、Islam等的模型都主要关注于任务特征的选择、模型的优化，忽视了预测时机的选择，往往假设完全获得了任务执行信息，然后进行预测。这导致理论研究和实际使用场景的不匹配。

2.3.2相关技术

任务失败预测使用了多变量时间序列分类模型，多变量时间序列的分类问题可以表述为：利用类别已知的多维时间序列训练样本建立分类模型，基于该模型对类别未知的样本进行类别预测。

2014 年Ge 等人[20] 提出了MCDCNN 模型专门用于多变量时间序列分类问题，考虑到每个变量对应的的时间序列内部可能含有不同的时序特征，它采用独立的卷积操作处理每一条时间序列。

2017 年Wang[8] 等人提出了Fully Convolutional Network(FCN)，该模型主要是由卷积网络组成，并且整个模型中没有使用局部的池化层。2017 年Wang[8] 等人提出了residual network (ResNet)，它是一种基于ResNet 结构的深度卷积网络，根据Fawaz 等人在2019 年的研究[21] 此模型是目前最优的多变量时间序列分类模型。

2.4负载预测问题相关研究及技术

2.4.1相关研究

目前关于trace 的研究大部分关注计算trace，I/O trace 方面则较少，尤其是基于开源I/O trace 的时序预测模型的研究则更加少见。

Gupta S[23] 等人在云平台CPU 资源使用率的预测问题上成功使用了长短期记忆单元，将CPU 使用率的预测问题建模为多变量时序预测问题，并且使用一个双向的LSTM 网络来捕捉过去对现在以及现在对过去这两方面的影响。

Di[24] 等人首先利用朴素贝叶斯模型预测了云环境下主机的CPU 和内存的平均使用率。他们使用观测窗口中的历史负载数据提取九个特征来对预测窗口中的均值负载进行预测。

Song 等人[25] 认为，云环境中负载的高方差是导致传统基于统计和基于机器的学习方法性能较差的主要原因。

Zhang 等人[26] 研究主要关注不同服务器监控指标对负载预测的影响，认为在监测指标种类很多时，很难确定某一指标是否有利于工作负载的预测，所以提出一个基于深度学习的负载预测模型，主要包括两个部分：监测指标选择模块和循环神经网络预测模块。

2.4.2相关技术

目前基于点值的负载预测算法大致可分为三种，基于统计的方法，基于传统机器学习的方法，基于深度学习的方法。基于统计的方法主要以ARIMA 为代表，主要使用回归模型描述一个平稳的时间序列；基于机器学习的方法将时间序列视为一个各个维度相互独立的向量，从而转化为一般回归问题；基于深度学习的方法，大多采用循环神经网络等序列模型，对序列元素中的依赖关系进行建模。

经典时间序列模型（例如AR、MA、ARMA 和ARIMA 等[27–30]）及其相应的分析方法是一种基于统计学的时间序列预测模型，表达式由公式2.1给出。在常见的时间序列模型中，ARIMA 模型能够很好的拟合不稳定的序列；对于平稳序列AIRMA 原本就有较好的拟合能力[22]。

基于机器学习的算法能提供更加优秀的非线性拟合能力。随着统计机器学习的发展，时间序列预测已经被定义为一个回归问题。

循环神经网络(RNN)[35] 是一种专门用于序列建模的深度神经网络，由于其在捕捉非线性关系方面的灵活性而受到广泛关注。然而，传统RNN 面临着梯度消失的问题，因此很难捕获序列的长期依赖。最近，克服了这一限制的长短时记忆单元(long short-term memory units, LSTM)[36] 在各种应用中取得了很大的成功。其组件结如图2所示。

基于特征增强的多任务LSTM负载转折点预测算法

3.1研究背景

随着人工智能的发展，基于机器学习和基于深度学习的服务器负载预测模型与资源管理策略紧密的结合在了一起。

在时间序列预测问题中，可以分为点值预测和趋势预测。点值预测问题即预测未来的某个或某些点的取值的问题，而趋势预测则有着更宽泛的定义，大致分为三类：预测下一个点相对于当前点是上升还是下降；预测下一个趋势片段的斜率和持续时间；预测下一个点是否是转折点。

目前对云环境下服务器负载预测的研究主要集中在点值预测。尽管机器学习和深度学习方法在基于点值预测的场景下取得了极大的发展，但是由于云环境下庞大的任务提交数量、繁多的任务类型，短时间内的服务器负载精确预测依然是一个严峻的挑战[41]。

对于点值预测，即使可以预测出未来的负载值，也很难知道服务器的内部状态和负载的动态变化趋势。相比点值的预测，趋势变化的预测有两点优势：一，不易受单点抖动噪声的影 ；二，蕴含更加丰富的长期状态信息。

本章针对Xia等人在2018 年的研究[10]的不足——所使用的特征大都是时间窗口内基础的统计特征，以及只基于预测点附近的单个时间窗口内的统计特征构建构型——研究云环境下服务器负载转折点预测算法。

3.2负载转折点预测问题建模

工作负载转折点的形式化定义如下：给定工作负载时间序列x =（x1; …x2; …xT ），T是工作负载时间序列的总长度，工作负载转折点是工作负载局部趋势逆转的点。在文献[12] 中，局部趋势往往是采用分段线性分割(或称表示) 算法[11](PLR) 来描述的。时间序列线性分割算法的目标是将给定的时间序列x 分割为一系列片段Xseg，其核心思想是在每一段中拟合一条直线，使总拟合误差最小。

本文认为，分割出来的段的边界只是转折点的候选集合，需根据双边的端点值大小行过滤操作，保证该点两侧趋势相反。

3.3算法框架



图7 基于特征增强的LSTM 负载转折点预测算法框架图

本文提出的服务器负载转折点预测算法框架如图7所示，它由两个部分组成：工作负载预处理模块和特征增强的多任务LSTM 预测模块。模型的输入是云环境下服务器的trace，可以是CPU、内存和其他工作负载组成的时间序列。

工作负载预处理模块负责从负载时间序列中提取波动性特征，以及为模型的训练生成标记数据，它们都被送入后续预测模块。预测模块采用本文提出的基于特征增强的多任务LSTM 模型（FEMT-LSTM），预测转折点并输出相应的预测标签。

3.4工作负载预处理

本文是基于监督学习模型对服务器负载转折点进行预测，训练监督学习模型需要监督数据，即对每个样本构建特征以及对应的标签。负载预处理模块需要从原始的负载时间序列中生成转折点标签，同时在观测窗口中提取时序波动特征。

3.4.1基于规则过滤的转折点标签生成

先按照3.2小节所述的定义采用分段线性分割法，自动生成转折点标签。由PLR 生成的段边界并不一定是转折点，所以需要进行无关点的滤除操作，本文提出了一种基于规则过滤的转折点标记算法，如算法1所示，该算法遵循的过滤规则如3.2节，公式3.1和3.2所示。

3.4.2 基于波动性的服务器负载时序特征提取

云环境下工作负载的时序特征通常是从具有固定大小的时间窗口中提取的，本文称此窗口为特征观测窗口，从该窗口提取的特征反映了不同的趋势变化。

本文使用的特征列于表1. 除了三个基础统计特征（当前工作负载，工作负载的平均值，工作负载的方差），本文提出了四个波动性特征，以反映观测窗口临近点的负载趋势情况。

3.5基于特征增强的多任务LSTM预测模型

3.5.1基于滑动窗口的序列式特征

对于每个时刻的原始负载xt，都可以采用3.4.2小节所述的方法在长度为n 的特征观测窗口内（xt-n+1; xt-n+2; …xt ） 提取对应的2 两种基础特征以及4 种波动特征。

3.5.2 多任务结构与特征增强特性

与原始负载序列x = (xt-w+1; xx-w+2…xt )2 Rw 不同，提取到的特征序列含有更加丰富的高层次的波动性含义。一种合理的做法是将特征序列和原始的负载序列分别建模。本文使用两个独立的LSTM 网络分别对原始负载序列和特征序列进行特征提取。

考虑到本文所提出的4 种波动性特征主要衡量的是负载的趋势以及变化的程度，这些特征在不同的原始负载的取值下可能代表着不同的含义，本文使用另外一个LSTM 层实现特征增强。

3.6实验验证

实验分为3部分：波动特征有效性实验；特征增强的多任务LSTM模型的有效性实验；模型简化实验。

3.6.1实验环境说明

实验环境见表2和表3。软件环境中TA-Lib[43] 和Tsfresh[44] 是两款时间序列分析工具，本文主要采用两者提供的特征计算函数进行特征提取，使用Keras 进行模型的搭建，模型的评价指标方面使用Scikit-learn[47] 中提供的函数进行计算。

3.6.2数据集

实验在开源的谷歌云服务集群trace 上进行。随机选择ID 为207776314（M1），908054（M2），1273805（M3）的三个节点作为本文的三个实验数据集；设置PLR 算法的分割阈值为0.0015 进行分割以及标记；按照时间顺序将每个数据集划分为训练集，验证集和测试集；前80％的数据用于训练，接着10％的数据用于验证参数，最后10％的数据用于测试。

3.6.3训练方法

本文使用minibatch 随机梯度下降和Adam[49] 优化器来训练模型。learning rate 设置为0.001。batch size 设置为64。参数通过标准反向传播来学习，损失函数为二值化交叉熵。

3.6.4评价指标

本文使用F1 值作为模型评估的主要指标，F1 值是由精确率（precision）和召回率（recall）计算出来的。precision 值和recall 均由真正例（TP）、假正例（FP）、假负例（FN）三种指标计算得到。

3.6.5对比算法

选取的基线模型如下：LR-Basic，LR-Fluctuant，WSVM-Basic，WSVM-Fluctuant，P-LSTM，S-LSTM。

三种深度学习模型：P-LSTM，S-LSTM 以及FEMT-LSTM 中使用的LSTM 隐层神经元个数均设置为256，特征观测窗口大小n 和WSVM-Fluctuant 设置为同一值，历史观测窗口w 设置为10 或5。所有模型均使用sklearn 和keras 实现。

3.6.6实验结果与结论

三个数据集上的对比结果如表4和图14所示。

基于通道注意力卷积模型的任务失败预测算法

基于长短期记忆网络的存储系统I/O负载短期时序预测算法

总结与展望

总结

展望

未来可能在以下方面进行优化：

（1）在工作负载转折点预测问题的研究中，Google trace提供的单个服务器的时序数据总长度不足10000，这种数据量对于基于深度学习的时间序列挖掘算法来说略显不足。最近有研究表明[60] 通过对抗学习（Adversarial Training）能够帮助模型从多条有差异的时间序列中学习到更加鲁棒的特征。后续可以研究基于多服务器负载数据的转折点预测算法。

（2））本文的第一和第二个工作都是基于谷歌集群进行的，而阿里巴巴在2018年发布了更大规模、更大数据量、更小监测粒度的trace数据。未来可以尝试探索该trace数据中的时序特征。