第 8 卷 第 3 期
2019 年 5 月
集 成 技 术
JOURNAL OF INTEGRATION TECHNOLOGY
Vol. 8 No. 3
May 2019

引文格式：
石婧文, 罗树添, 叶可江, 等. 电商集群的流量预测与不确定性区间估计 [J]. 集成技术, 2019, 8(3): 55-65.
Shi JW, Luo ST, Ye KJ, et al. Traffic prediction and uncertainty interval estimation for e-commerce clusters [J]. Journal of Integration Technology, 2019, 8(3): 55-65.

电商集群的流量预测与不确定性区间估计

石婧文[1,2] 罗树添[1,2] 叶可江[1] 须成忠[1]

[1](中国科学院深圳先进技术研究院 深圳 518055)
[2](中国科学院大学 北京 100049)

摘 要 流量预测对智能容量规划和任务调度具有重要意义，然而大规模电商集群的流量会出现各种不确定的突发事件，如线上促销活动、用户聚集请求等。这些不确定性事件会导致时间序列中出现很多突发脉冲，从而给流量预测带来巨大挑战。同时，容量预测应当对不确定性具有鲁棒性，即能很好地应对未来可能出现的情况，保证集群稳定性，而并非严格地根据预测值进行容量收缩。针对大规模分布式电商集群的流量场景以及动态容量规划的需求，该文提出了包含不确定性估计的流量实时预测框架。该框架基于多变量的长短期记忆网络自动编码器和贝叶斯理论，在进行流量确定性预测的同时能够给出准确的不确定性区间估计。

关键词 电商流量；时间序列预测；长短时记忆神经网络；不确定性估计
中图分类号 TP 181 **文献标志码** A doi: 10.12146/j.issn.2095-3135.20180325001

Traffic Prediction and Uncertainty Interval Estimation for E-Commerce Clusters

SHI Jingwen[1,2] LUO Shutian[1,2] YE Kejiang[1] XU Chengzhong[1]

[1](*Shenzhen Institutes of Advanced Technology*, *Chinese Academy of Sciences*, *Shenzhen* 518055, *China*)
[2](*University of Chinese Academy of Sciences*, *Beijing* 100049, *China*)

Abstract Traffic prediction is of great significance for intelligent capacity planning and task scheduling. However, large-scale e-commerce cluster traffics have various uncertain emergencies, such as online promotion activities and user aggregation requests. These uncertain events may cause many bursts in the time series, which poses a huge challenge to traffic prediction. At the same time, capacity prediction should be robust to uncertainty. That is, it should cope well with possible future situations and ensure cluster stability, rather

收稿日期：2019-03-25 **修回日期**：2019-04-16
基金项目：科技部 973 项目(2015CB352400)；国家自然科学基金项目(61702492)；装备预研项目(61400020403)；深圳市学科布局项目(JCYJ20170818153016513)；深圳市自由探索项目(JCYJ20170307164747920)
作者简介：石婧文，硕士研究生，研究方向为分布式系统与机器学习；罗树添，博士研究生，研究方向为分布式系统与机器学习；叶可江，博士，副研究员，硕士研究生导师，研究方向为云计算、大数据系统及网络；须成忠(通讯作者)，博士，教授，博士研究生导师，研究方向为分布式系统、互联网与云计算、高性能计算、无线嵌入式系统等，E-mail：cz.xu@siat.ac.cn。

than shrink the capacity strictly based on the prediction. For the traffic scenarios of large-scale distributed e-commerce clusters and the requirements of dynamic capacity planning, this paper proposes a real-time load forecasting framework with uncertainty estimates. The framework is based on multivariate long short-term memory auto-encoder and Bayesian theory, which can provide accurate uncertainty interval estimation while performing flow deterministic prediction.

Keywords e-commerce traffic; time-series prediction; long-term memory neural network; uncertainty estimation

1 引 言

随着越来越多企业将服务与产品迁移到云计算集群，提高数据中心的资源利用率成为云服务提供商降低运营成本的关键手段。然而大量集群运行数据表明，即使采用虚拟化技术进行负载整合，数据中心的平均中央处理器(Central Processing Unit，CPU)利用率依旧难以超越50%[1-3]。为了提高资源利用率，弹性扩容、在线与离线任务混布调度、虚拟机协同部署等多种资源管理技术应运而生。而预测模型可帮助这些资源管理技术预先判断，做出更好的决策。Zhu 等[4]也指出未来云资源管理应当是及时响应与预测相结合进行决策。但目前用来进行集群流量预测的算法都没有考虑不确定性，而在生产型云集群中稳定性是一切的基础，资源利用率的提升必须首先保证集群稳定性。因此，预测算法对未来不确定性的估计与预测精确度同样重要，甚至更为重要。

传统的时间序列统计模型往往有一定的前提条件要求，如前向神经网络和自回归滑动平均模型(Auto-Regressive and Moving Average Model，ARMA)要求数据的 N 阶差分具有平稳性[5]。而实际场景中(如电商场景)，流量存在很多突发脉冲，它们严重破坏了时序数据的平稳性。另外，很多非机器学习的时间序列模型不能直接进行多变量输入，如 Facebook 在 2018 年开源出来的 Prophet 算法[6]。而神经网络不需要输入数据满足任何假设，可以很好地捕捉非线性特征，直接接受多变量输入，利用多维特征进行训练。近几年，神经网络在交通、降雨量等时间序列预测问题的研究上取得了很好的结果[7-8]，尤其是递归神经网络(Recurrent Neural Network，RNN)中的长短期记忆(Long Short-Term Memory，LSTM)网络、门控循环单元(Gated Recurrent Units)等。

本文提出了基于 LSTM 神经网络进行流量鲁棒预测的算法，对不确定性进行估计，以保证决策安全。由于数据噪声和网络参数的共同作用，神经网络模型中存在 3 种不确定性，即模型认知不确定性(Epistemic Uncertainty)、同方差不确定性(Homoscedastic Uncertainty)和异方差不确定性(Heteroscedastic Uncertainty)。Zhu 等[9]利用蒙特卡洛 Dropout 和自动编码机对 3 种不确定性进行估计，但在数据集有限情况下，同方差不确定性存在严重的过度估计问题。Kendall 和 Gal[10]通过损失函数预测异方差不确定性，并进一步给出了异方差不确定性和模型认知不确定性的结合方法，但未给出同方差不确定性的估计方法。针对以上不足，本文提出的在线流量预测与不确定性估计模型框架，考虑了实际大规模分布式电商集群的流量特点与弹性资源调度、动态容量规划的需求。实验结果表明，本文提出的方法具有以下优点：(1)适合实时环境；(2)提出一种新的

同方差不确定性估计方法，并将 3 种不确定性方法进行结合，很好地解决了不确定性区间估计的挑战，并且本文提出的不确定性估计方法可结合不同神经网络结构应用于不同场景的置信区间估计；(3)对异常情况体现出良好的鲁棒性。

2　相关工作

近年来，有很多工作对流量预测与资源管理的结合进行了研究，预测方法可分为基于排队论、经典时间序列分解和机器学习三类。Urgaonkar 等[11]和 Bennani 等[12]利用排队网络来模拟集群应用接收、处理请求的过程。Urgaonkar 等[11]对多层架构应用进行建模并考虑了缓存、并发等因素，但目前大规模应用多采用微服务架构，微服务将大的应用拆分为小的服务，服务间依赖关系形成复杂的图结构，很难进行准确地排队模拟建模。Bennani 等[12]对同一应用多种负载建立排队论模型，取得较好结果。但排队论需要请求到达符合泊松分布特点，实验发现在实际生产中，每天的电商应用请求呈现长尾、双峰、重尾等分布特点，不能通过卡方检验，因此不符合泊松分布条件。Meng 等[13]利用 ARMA 模型对负载时间序列进行预测，并用相关矩阵实现不同虚拟机在物理机器上的组合部署优化和容量估计。ARMA 算法每次预测需要回顾所有历史数据，由于虚拟机部署不需要实时优化，可以使用预测时间开销大的算法，但动态资源管理要求算法必须能实时预测。研究人员[14-15]探索了机器学习预测方法提升资源利用率。Baughman 等[15]利用多媒体、网页浏览、历史日志、机器日志等进行机器学习预测，为美国高尔夫与羽毛球联赛服务的私有云提供资源预测，然而收集与处理如此大量信息是很困难的。我们希望借助流量与时间等基本维度信息进行预测，通过不确定性估计弥补无法收集应用海量信息的限制，降低信息收集与处理代价，使模型更好地适用于不同应用。

3　数据集与特征工程

本文通过分布式链路追踪技术对集群机器进行日志埋点，并对集群流量进行分钟级别采样，其中集群涵盖 1 000 多台容器。本文将原始数据分为 3 个子集，3 份数据集的总时间跨度超过 100 天，每份数据包含 40 天的训练时序和随后 7 天的测试时序，测试覆盖 21 天。数据集第一、二周为正常情况下的流量数据；最后一份数据时间序列总体趋势快速下降，预测数据发生明显模式变化以验证算法在异常情况下的健壮性。

原始数据集由于受网络传输失败等因素影响，数据缺失率严重(2%～20%)；同时，时间序列平稳性被数据抖动严重破坏，尤其在高峰期不确定性脉冲大大增多。另外，不同于气象、交通等场景，可以从传感器收集到高维密切关联的数据，电商流量预测可以依赖的属性并不丰富。因此，需要对数据进行预处理和属性扩展。

在特征工程中，本文进行了以下预处理操作：对时间序列进行平滑操作，选择平滑窗口内最大值保障生产型云集群决策的鲁棒性。该操作改善了数据抖动问题，有助于提高神经网络训练效果，并将数据缺失缩小至 0，避免了填充算法引入新的噪声。本文使用滑动方式生成近似时间序列，并将时间窗口大小设为 5 min，主要基于以下考虑：(1)采用 5 min 时间窗处理后的时间序列与原始时间序列几乎重合，不损失有用特征；(2)5 min 时间窗符合资源调整的粒度，可以保障决策灵敏性。例如，目前应用广泛的调度框架 Kubernetes，其默认资源动态调整的间隔就为 5 min[16]。而在实际生产中，由于网络传输、资源变更前期准备等因素会带来额外的时间延迟，资源调整所需的时间远大于 5 min。平滑处理后，本文对数据进行标准化(映射到均值为 0、方

差为 1 的空间中)。此外，本文加入星期、时刻(分钟)等时间维度信息辅助预测算法。

4 研究方法

大规模分布式电商集群的流量预测主要有以下难点：(1)电商系统流量受到促销活动、机器刷单等影响，容易出现突发性脉冲，而很多基于平稳性假设的时间序列预测模型难以捕获这些脉冲，如自回归积分移动平均模型(Autoregressive Integrated Moving Average Model，ARIMA)。(2)脉冲具有不确定性，难以准确预测。例如，促销活动带来的流量不确定性与促销力度、促销方式、用户心理有关，难以获得所有相关信息；有些应用会因为机器刷单在特定时间段产生流量，但在时刻上不是每天严格对齐，这带来了不确定性；用户活跃的时间段(早上 8 点到晚上 12 点)会产生很多时刻与幅度随机变化的突发脉冲，而用户行为同样具有很大不确定性。因此，本文将突发性脉冲预测转化为不确定性区间的预测。(3)除了经济活动带来的挑战，集群限流、集群故障也会导致时间序列出现异常波动。为了保障集群稳定运行，预测模型需要对异常情况具有一定鲁棒性。本文为电商集群的资源管理、容量规划决策提供预测模型，除了考虑电商集群流量本身的复杂性，还需要满足动态资源管理的需求。(4)动态调整后的容量需要承载未来一段时间内可能出现的最大负载，从而保证用户服务质量，因此预测模型不仅要提供流量的预测值，还需要估计出最坏情况(最大流量)。而对于有些模型(如神经网络)，不确定性的估计本身就是一项很有挑战的任务。传统统计方法中的置信度区间取决于数据波动而与统计方法无关，而神经网络中的不确定性还可能来源于模型参数，即模型本身的认知存在不确定性。(5)模型需要具备在线快速预测的能力，在离线分析中表现好的模型可能在在线场景中并不能发挥作用。

在本文中，首先利用基于 LSTM 的 Seq2Seq 神经网络来满足对时序模型提出的准确度与预测时间的要求；然后，利用贝叶斯理论为网络加入不确定性区间估计，以应对不确定性带来的挑战。

4.1 Seq2Seq 神经网络

为了解决时序预测问题，Cho 等[17]在 2014 年提出了 Seq2Seq 编码-解码网络结构，该模型提出后受到了持续关注。随后，Sutskever 等[18]在同年提出了利用 LSTM 作为基本单元的 Seq2Seq 网络，进一步解决了长时依赖问题。其中，Seq2Seq 是一种神经网络压缩算法，它能将输入序列的信息映射到低维隐层空间，解码器通过隐层间信息还原高维输出，且 Seq2Seq 允许输入输出为不同长度的序列。本文利用 LSTM 作为 Seq2Seq 的编码与解码器基本单元，具体网络结构如图 1 所示。LSTM 是递归神经网络结构的一种，每个单元是一个细胞，类似于神经元。但与全连接层神经元最大的不同之处在于，LSTM 单元拥有 3 个门(遗忘门、输入门、输出门)和 2 个状态(隐状态和细胞状态)。LSTM 利用细胞状态携带着信息在不同网络层或不同时间步之间流动。LSTM 的门是一组函数，由乘法、加法和激活函数组成，它们共同控制着 LSTM 单元的状态与输出。Seq2Seq 结构网络不仅解决了输入、输出长度不相等的问题，还因其网络层数较浅，能够达到实时预测要求的速度，从而更加适合动态资源调整场景。

4.2 不确定性区间估计

贝叶斯神经网络存在 3 种不确定性：模型认知不确定性、同方差不确定性和异方差不确定性。模型认知不确定性由网络参数造成，会随着数据集增大被解释，即模型不确定的认知逐渐减小；而同方差不确定性和异方差不确定性来自于数据集。其中，同方差不确定性存在不同数据中；异方差不确定性会随着输入的改变而改变。

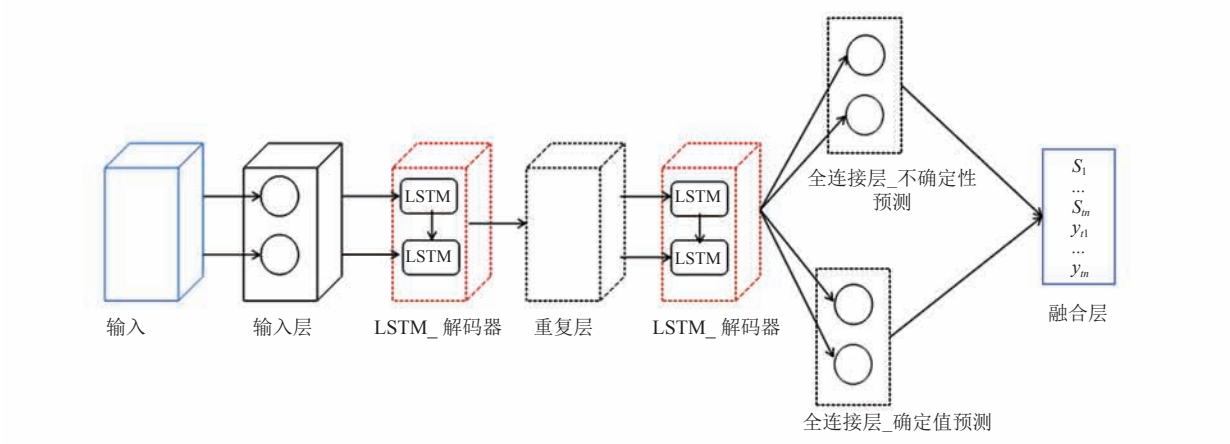


图 1　Seq2Seq 结构的鲁棒性预测网络

Fig. 1　Robust prediction network of Seq2Seq structure

普通神经网络是点优化问题，模型经过训练找到最优解或局部最优解，而贝叶斯神经网络认为网络参数和偏置服从一定分布，然后利用贝叶斯公式将不确定性估计加入预测。

(1) 记模型认知不确定性为 u_1。蒙特卡洛 Dropout 是模型认知不确定性常用的近似推理方法[9-10]。本文对相同输入进行 N 次预测，每次用蒙特卡洛 Dropout 方法对网络权重 W 进行抽样，可以得到 N 个预测值 y。此时 y 的不确定性是由网络权重引起的，因此输出 y 的方差即为模型认知不确定性平方 u_1^2。由公式(1)可得到 y 的方差。

$$u_1^2=\frac{1}{N}\sum_{n=1}^{N}f^{W_n}(x)^{\mathrm{T}}f^{W_n}(x)-E(y)^{\mathrm{T}}E(y) \tag{1}$$

其中，W_n 为第 n 次运行的网络权值矩阵；$f(x)$ 为每次预测的结果；$E(y)$ 为 N 次预测的期望；T 为矩阵转置。

(2) 记异方差不确定性为 u_2。Kendall 等[10]利用贝叶斯理论、最大似然估计(Maximum Likelihood Estimation，MLE)和最大后验估计(Maximum a Posteriori Estimation，MAP)推理证明公式(2)可以估计出异方差不确定性。本文以公式(2)作为新的损失函数，将 u_2 变为可预测项作为网络部分输出。Kendall 等[10]将预测得到的异方差不确定性与模型认知不确定性结合，成功运用到图像像素回归任务中。本文利用论文中推导出的公式(2)作为神经网络的误差函数(Loss Function)，对时间序列中的异方差不确定性进行预测。

$$Loss=\frac{1}{D}\sum_i\frac{1}{2}\exp(-s_i)\left\|y_i-\hat{y}_i\right\|^2+\frac{1}{2}s_i \tag{2}$$

$$s_i=\log u_2^2(x_i) \tag{3}$$

其中，D 是预测时间窗长度；s_i 是 u_2 的函数；y_i 和 $\hat{y}_i$ 分别是时刻 i 的真实值与预测值。同时由公式(3)可知，此时异方差不确定性 u_2 是输入 x_i 的因变量。

(3) 记同方差不确定性为 u_3。根据 Zhu 和 Laptev 等[9]的推导，一个利用趋于无限数据集合训练的模型(模型认知不确定性趋于零)，保持权重矩阵不变，在独立验证集(独立于训练集合)上预测与真实值的误差 *var*1 可以看作是同方差不确定性的渐进无偏估计。但由于训练集合有限，这种计算方式明显会带来不确定性的过度估计。因此对于有限数据训练出来的模型，本文从训练集中抽样出一个验证集合，用该集合预测与真实值的误差 *var*2 作为同方差的估计量。*var*2 比独立验证集上的估计量 *var*1 要小，由于训练集合进行过迭代训练，模型认知不确定性大大减

小(但 *var*2 不一定比真实的同方差不确定性小，因为模型认知不确定性无法从 *var*2 中分离)。本文在实验中发现不确定性 u_1+u_2 的 95% 置信度区间效果在 96%～99%，因此，*var*2 比 Zhu 和 Laptev 等[9]提出的 *var*1 更能缓解前两种不确定性的估计过度，避免了区间过度发散。同时，由于 *var*2 中包含的模型认知不确定性与时刻 *t*(一天内第几分钟)有关：有些时刻时序一直很稳定，模型认知不确定性较小，得到的 *var*2 也相应地会比较小；反之，经常出现秒杀、抢券等促销活动的时刻，模型难以进行准确预测，因此模型认知不确定性大，得到的 *var*2 也相应增大。所以，计算同方差不确定性时，本文考虑了时刻差异，在训练集合中同预测时序的时刻信息一致的数据中随机抽样，构成最终的验证集。

$$u_3^2=\frac{1}{M}\sum_{m=1}^{M}(\hat{y}_t-y_{t\_true})^2 \tag{4}$$

其中，*M* 为验证集 *t* 时刻的抽样总数。

(4)本文利用公式(5)对 3 种不确定性影响叠加得到 *var*。*var* 与预测值 $\hat{y}$ 结合可得到置信度区间，即 $\hat{y}_i \pm z_{\alpha/2}\times var$，其中 α 为显著性水平。本文取 95% 置信度区间(显著性水平为 0.05)进行区间计算，此时，$z_{\alpha/2}$ 约等于 1.96。

$$var=\sqrt{{u_1}^2+{u_2}^2+{u_3}^2} \tag{5}$$

5 实验结果

本文算法的实验运行环境为 Intel Core i5，1.6 GHz，操作系统 macOs 10.13. 数据处理与模型由 Python2.7+Keras2.2.4+Tensorflow1.12.0+sklearn0.0 实现。

5.1 实验模型与参数调节

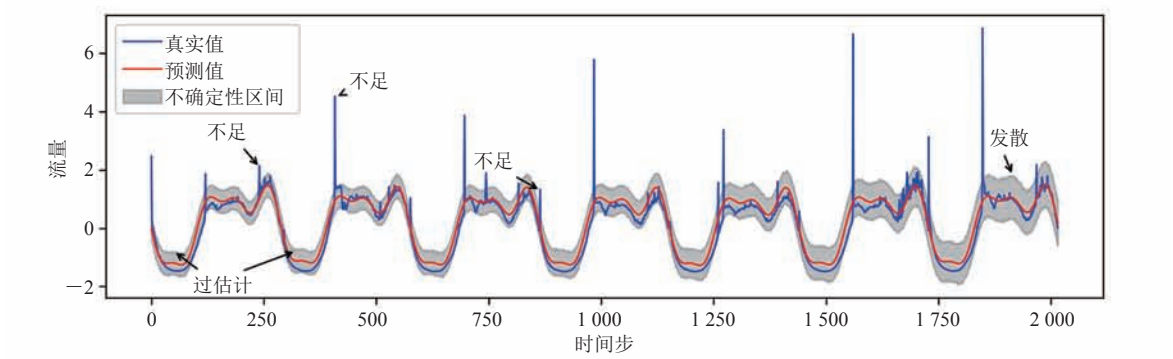
本文训练了另外 4 个不同的回归预测模型进行对照实验，用来评估提出的新神经网络模型(UN)对未来流量的预测效果，表 1 给出了 5 个模型的介绍。

实验过程中，所有神经网络采用相同深度，并保持结构近似。UN 与 Seq2Seq 网络包含两个 LSTM 层和一个全连接层(Fully Connected Layer)，两个 LSTM 层在非循环步使用 0.2 的比例进行 dropout。FC 网络包含 3 个全连接层，同时在两个全连接隐层前加入比率为 0.2 的两个 dropout 层，使用修正线性单元(Rectified Linear Unit，ReLU)作为激活函数。在实践中，本文发现 UN 网络在每层 LSTM 单元数为 32 时就可以达到最佳效果，而 Seq2Seq 和 FC 网络在每层单元数为 64 时效果更好。因此，在对照实验结果中，本文提出的网络使用了对照神经网络模型一半的单元数目。另外，如公式(2)所示，由于本文在提出的 UN 模型中使用了二阶损失函数，因此在 Seq2Seq 和 FC 网络中，本文使用了二阶函数均方误差(Mean Square Error，MSE)作为损失函数。所有神经网络模型训练最大周期为 500，采用 Adam 优化器，网络收敛时可提前结束训练(EarlyStopping)，EarlyStopping 不仅可以缩短训练时间，还可以增强网络泛化能力，缓解过拟合

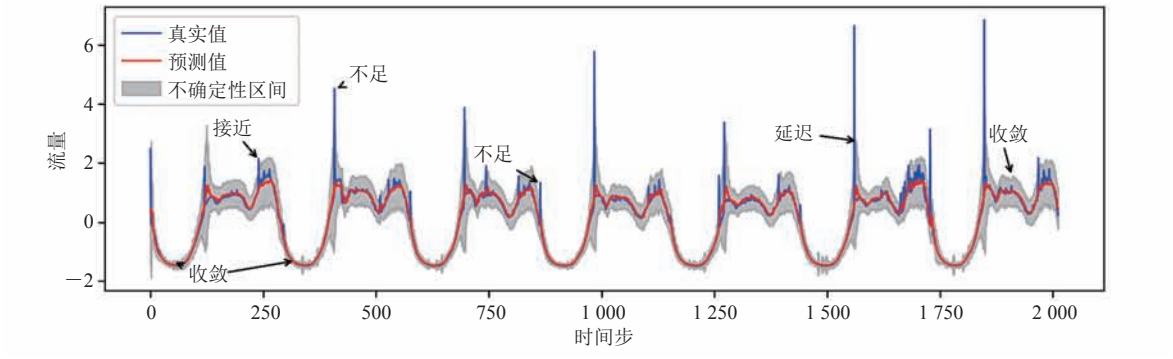
表 1 实验模型说明

Table 1 Experimental model description

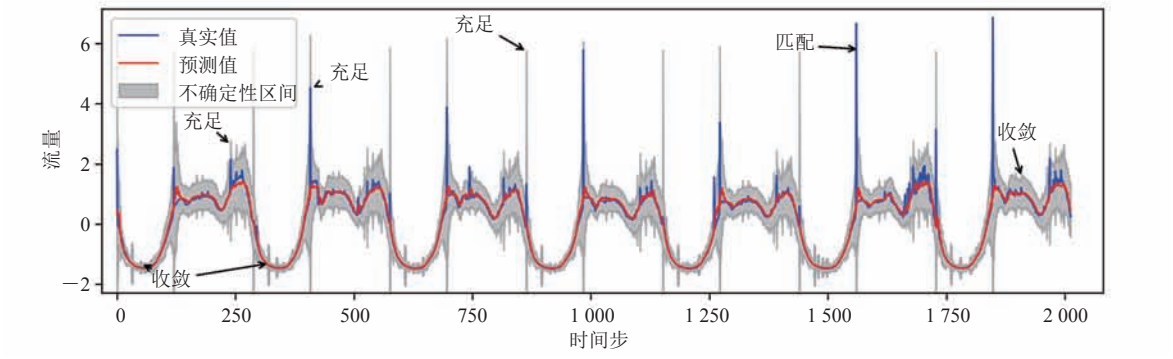
模型名称	模型说明
UN	本文提出的不确定性神经网络模型（UN2 计算了两种不确定性，UN3 计算了三种不确定性）
Seq2Seq	Encoder-Decoder 结构的神经网络，本文采用的原始网络结构（基本单元为长短期记忆网络）
FC	全连接神经网络，与本文网络层数相同（dropout 层不计入）
Prophet	Facebook 的开源时间序列预测算法，带有置信度区间估计
SVR	支持向量机回归预测模型



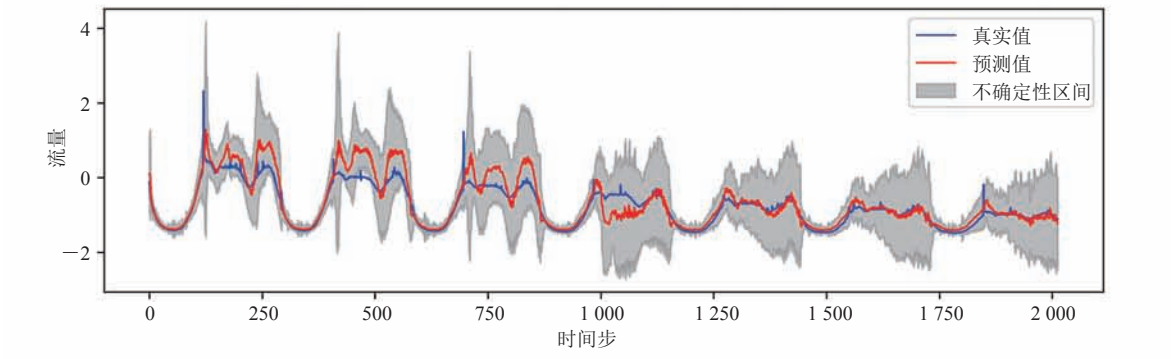
(a) Prophet



(b) UN2



(c) UN3



(d) 异常星期

图 2　区间预测结果

Fig. 2　Interval prediction result

问题。

本文使用最近 3 h 事件窗口的时间序列作为机器学习算法输入，未来 15 min 时间窗口作为输出。在训练 Prophet 时需要输入所有历史数据，分析长期的时间规律，预测时指定预测时间窗口长度。由于 Prophet 算法实时性不高，因此本文指定未来 7 天的时间窗口作为输出。其中，Prophet 算法自带参数调节功能，另外 4 个机器学习算法使用 GirdSearch 进行参数调节。

5.2 不确定性区间预测对照评估

在第 5.2.1 小节中，图 2(a)～(d)展示了 Facebook 的开源算法 Prophet 和本文算法的不确定性区间的预测效果。在第 5.2.2 小节中，本文使用 3 个指标量化不确定性区间估计的效果好坏，并进行对照、自身对照实验。

5.2.1 不确定性区间效果图

图 2 为 Facebook 开源算法和 UN 的预测效果，其中横坐标每个时间步代表 5 min，纵坐标为流量标准化后的值。本文将结合 5.2.2 小节对相应结果进行分析。

5.2.2 不确定性区间评估指标

为了定量评估不确定性区间预测效果的好坏，本文设计了 3 个指标：覆盖率、均摊覆盖率和离群点平均距离。

(1)区间覆盖率：区间覆盖真实值点的百分比，区间的覆盖率越大，覆盖的点越全面。由表 2 可知，本文提出的 UN3 模型在第一周、第二周预测中覆盖率分别比 UN2 高 1.1% 和 1%，并比 Prophet 高 2.57% 和 4.66%，在第三周异常情况预测中更比 Prophet 高 67.4%。如图 2(a)～(c)所示，本文模型较 Prophet 能获得更优覆盖率，主要是因为 Prophet 算法虽然能够对趋势进行较好估计，但对于脉冲几乎不具备预测和区间估计能力，而本文模型克服了这一点。UN3 较 UN2 能获得更优覆盖率的主要原因是：①UN3 在规律脉冲处预测值更高，而 UN2 很多脉冲超出了估计区间；②UN2 对脉冲的估计可能发生错位、延迟，而加入考虑了时间差异的同方差不确定性后，UN3 也纠正了这一点。

表 2 区间覆盖率

Table 2 Interval coverage

模型	区间覆盖率（%）		
	第一周（正常）	第二周（正常）	第三周（异常）
Prophet	97.123 015 87	95.089 285 71	30.803 571 43
UN2	98.609 041 23	98.758 072 53	96.423 248 88
UN3	99.701 937 41	99.751 614 50	98.211 600 00

(2)均摊覆盖率：在覆盖率相同的情况下，覆盖面积越小越好。本文利用均摊覆盖率表示单位面积分摊的覆盖率。覆盖面积由对区间上下界差值累加得到。如表 3 所示，Prophet 算法只有在第二周时比本文模型 UN3 高 0.001 8(占 UN3 的 4%)，而本文模型 UN3 在第一周、第三周时分别比 Prophet 高 0.006 4 和 0.014 6(占 Prophet 的 17%、81.2%)。UN2 三周比 Prophet 分别高 0.012 6、0.008 5、0.011 2(占 Prophet 的 33.4%、18.3%、61.1%)。UN2 与 UN3 效果好于 Prophet 算法的主要原因是：①如图 2(c)和表 3 所示，日常平稳的波谷处 UN3 给出的不确定性区间更加收敛；②UN3 的区间估计不会随时间的推移而越来越发散。

表 3 均摊覆盖率

Table 3 Average coverage

模型	均摊覆盖率（%）		
	第一周（正常）	第二周（正常）	第三周（异常）
Prophet	0.037 610 96	0.046 617 07	0.018 007 51
UN2	0.050 185 54	0.055 159 04	0.029 229 81
UN3	0.044 043 90	0.044 828 48	0.032 617 00

(3)离群点平均距离：对于区间没有包含的点，区间边界距离这些点的距离越小，区间的参考价值越大。离群点平均距离是区间外的点到边界距离的平均值。如表 4 所示，Prophet、UN2 的离群点平均距离是本文模型 UN3 的 1.43～68.86 倍。说明对于区间外未被覆盖到的点，本文模型计算的区间边界距离这些点会更加

接近。本文模型在均摊覆盖率上的轻微损失是必要的，这样可以帮助本文模型更加接近区间外的点，保障决策安全性。

表 4　离群点平均距离

Table 4　Average outlier distance

模型	平均距离		
	第一周（正常）	第二周（正常）	第三周（异常）
Prophet	0.023 744 0	0.019 333 6	0.187 311 7
UN2	0.012 087 0	0.012 353 5	0.003 932 3
UN3	0.005 247 4	0.000 810 0	0.002 720 0

5.3　准确值预测对照评估

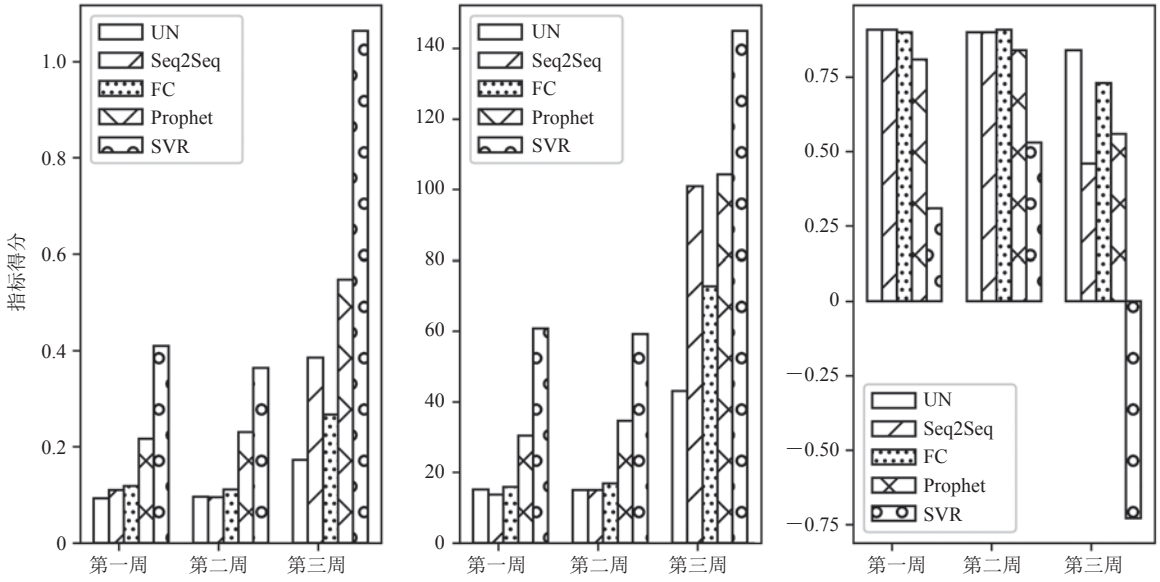
预测准确值的结果采用平均绝对值误差(Mean Absolute Error，MAE)、对称平均绝对百分比误差(Symmetric Mean Absolute Percentage Error，SMAPE)、R^2 三个度量指标。

(1) *MAE* 代表预测值与真实值的绝对值接近程度。该指标越小，则模型预测效果越理想，如公式(6)。图 3(a)给出各个模型的 *MAE* 值，本文提出的算法在给出不确定性区间估计的同时没有牺牲预测的准确度，在三周的预测上均获得了最佳效果(第一、二、三周的 *MAE* 值依次为 0.092、0.095、0.172)，比起相似结构的 Seq2Seq 甚至有所提升。在第三周异常情况下，有些模型错误率提升很大，如 Seq2Seq 提升了 305.8%、SVR 模型提升了 193.1%。本文所提方法的错误率提升了 81.5%，显示出对异常情况较强的鲁棒性。

$$MAE=\frac{1}{n}\sum_{i=1}^{n}\left|(y_i-\hat{y}_i)\right| \tag{6}$$

(2) *SMAPE* 指标考虑了预测与真实值量级的差异性，是时间序列预测常用的评价指标[19]。*SMAPE* 越小也代表模型效果越好，如公式(7)所示。由图 2 的比较可得，本文提出的模型在 3 个星期的预测中，*SMAPE* 错误率表现为最低或次低，UN3 前两周的 *SMAPE* 错误率与 Seq2Seq、FC 模型相差 2% 以内，第三周 Seq2Seq、FC 分别比 UN3 高 134.4%、68.8%，Prophet 则分别比 UN3 高 100.3%、132.2%、142.2%(相对于 UN3 的比例)。

$$SMAPE=\frac{100\%}{n}\sum_{i=1}^{n}\frac{\left|\hat{y}_i-y_i\right|}{(\left|\hat{y}_i\right|+\left|y_i\right|)/2} \tag{7}$$



(a) *MAE*　　(b) *SMAPE*　　(c) R^2

图 3　精确预测定量评估

Fig. 3　Quantitative assessment of precise prediction

(3)R^2 指标如公式(8)所示，该指标越接近1，代表模型效果越好，而糟糕模型的 R^2 得分则为负。由图 3 可得，本文所提出的模型在 R^2 得分上第一周为最优，第二周为次优且仅比最高 FC 低 1%，第三周高于 Seq2Seq 82.6%、FC 15.1%。另外，尽管 Prophet 在 *SMAPE* 和 *MAE* 误差上的表现不佳，但 R^2 指标在日常得分较高，说明 Prophet 在总体趋势预测上取得了一定效果。而预测效果很差的模型，R^2 指标会给出负值来体现模型糟糕程度，如 SVR 为−0.73。

$$R^2=1-\frac{\sum_{i=1}^{n}(y_i-\hat{y}_i)^2}{\sum_{i=1}^{n}(y_i-\overline{y}_i)^2} \tag{8}$$

6 结 论

本文提出了一种面向大规模电商的流量鲁棒预测解决方案，给出可以同时进行确定值预测与不确定性区间估计的流量预测网络模型。结果表明，在依赖信息很少，时间序列平稳性被不确定性突发脉冲严重破坏情况下，该模型能对日常流量和不确定性进行很好的预测，能给出准确的不确定性区间，同时整个模型对异常情景有很好的鲁棒性。本文提出的不确定性区间估计方法可以灵活地结合不同神经网络结构，因此可以进一步探索更复杂的时序预测神经网络。未来将尝试为 Seq2Seq 网络添加注意力机制，并将时序神经网路与经典信号分解(如傅立叶分析、小波理论等)进行结合，进一步增强网络特征提取能力。模型对整体变化趋势的异常时间具有较强鲁棒性，虽然精准预测还不能很准确，但不确定性区间可以保障资源管理安全性。未来需要针对临时大促、集群剧烈变化等极端事件进一步优化，可选择更多特征、建立专门的特征提取网络或预测网络。

本文所提出的流量预测模型可进一步与其他模型结合，如单容器负载能力预测模型，帮助 Kubernetes 等资源管理系统实现安全的容量扩缩。另外，不确定性区间能够帮助检测超出置信度区间的异常峰值。

参 考 文 献

[1] Reiss C, Tumanov A, Ganger GR, et al. Towards understanding heterogeneous clouds at scale: google trace analysis [J]. Intel Science and Technology Center for Cloud Computing, 2012: 84.

[2] Shan Y, Huang Y, Chen Y, et al. LegoOS: a disseminated, distributed {OS} for hardware resource disaggregation [C] // 13th Symposium on Operating Systems Design and Implementation, 2018: 69-87.

[3] Lu C, Ye K, Xu G, et al. Imbalance in the cloud: an analysis on alibaba cluster trace [C] // 2017 IEEE International Conference on Big Data, 2017: 2884-2892.

[4] Zhu X, Young D, Watson BJ, et al. Integrated capacity and workload management for the next generation data center [C] // Proceedings of the 5th International Conference on Autonomic Computing, 2008.

[5] Makridakis S, Hibon M. ARMA models and the Box-Jenkins methodology [J]. Journal of Forecasting, 1997, 16(3): 147-163.

[6] Taylor SJ, Letham B. Forecasting at scale [J]. The American Statistician, 2018, 72(1): 37-45.

[7] Zhao Z, Chen W, Wu X, et al. LSTM network: a deep learning approach for short-term traffic forecast [J]. IET Intelligent Transport Systems, 2017, 11(2): 68-75.

[8] Shi XJ, Chen ZR, Wang H, et al. Convolutional LSTM network: a machine learning approach for precipitation nowcasting [C] // Advances in Neural Information Processing Systems, 2015: 802-810.

[9] Zhu L, Laptev N. Deep and confident prediction for time series at uber [C] // 2017 IEEE International Conference on Data Mining Workshops (ICDMW), 2017: 103-110.

[10] Kendall A, Gal Y. What uncertainties do we need in bayesian deep learning for computer vision? [C] // Advances in Neural Information Processing

Systems, 2017: 5574-5584.

[11] Urgaonkar B, Pacifici G, Shenoy P, et al. An analytical model for multi-tier internet services and its applications [J]. ACM SIGMETRICS Performance Evaluation Review, 2005, 33(1): 291-302.

[12] Bennani MN, Menasce DA. Resource allocation for autonomic data centers using analytic performance models [C] // Second International Conference on Autonomic Computing, 2005: 229-240.

[13] Meng X, Isci C, Kephart J, et al. Efficient resource provisioning in compute clouds via VM multiplexing [C] // International Conference on Autonomic Computing, 2010.

[14] Shen Z, Subbiah S, Gu X, et al. Cloudscale: elastic resource scaling for multi-tenant cloud systems [C] // Proceedings of the 2nd ACM Symposium on Cloud Computing, 2011: 5.

[15] Baughman AK, Bogdany RJ, McAvoy C, et al. Predictive cloud computing with big data: professional golf and tennis forecasting [J]. IEEE Computational Intelligence Magazine, 2015, 10(3): 62-76.

[16] The Kubernetes Authors. Horizontal pod autoscaler [EB/OL]. [2019-04-16]. https://kubernetes.io/docs/tasks/run-application/horizontal-pod-autoscale/.

[17] Cho K, Van Merriënboer B, Gulcehre C, et al. Learning phrase representations using RNN encoder-decoder for statistical machine translation [J]. arXiv: 1406.1078, 2014.

[18] Sutskever I, Vinyals O, Le QV. Sequence to sequence learning with neural networks [C] // Advances in Neural Information Processing Systems, 2014: 3104-3112.

[19] Laptev N, Yosinski J, Li LE, et al. Time-series extreme event forecasting with neural networks at uber [C] // International Conference on Machine Learning, 2017: 1-5.