在学位项目信息和通信技术方面，

第二个周期,30学分

,

使用长时间的时间数据异常检测。

内存(LSTM)

阿卡什辛格



TRITA TRITA-ICT-EX-2017:124



利用长时间对时态数据进行异常检测。

短期记忆(LSTM)

阿卡什辛格

硕士学位论文的主要内容是KTH信息和通信技术。

导师:丹尼尔·吉布莱德主考官:马格纳斯·博曼。

工业主管:Mona Matti, Rickard Coster (Ericsson)

TRITA-ICT-EX-2017:124

文摘

我们探讨了利用长短时记忆(LSTM)对时间数据进行异常检测的方法。由于获得标记异常数据集的挑战，采用了无监督的方法。我们用LSTM单元训练递归神经网络(RNNs)，学习正常的时间序列模式并预测未来的值。所得到的预测误差被建模为给出异常分数。我们研究了保持LSTM状态的不同方法，以及使用固定数量的时间步对LSTM预测和检测性能的影响。并将LSTMs与具有固定大小时间窗的前馈神经网络进行了比较。我们的实验，有三个真实的数据集，表明虽然LSTM RNNs适用于一般的时间序列建模和异常检测，但是保持LSTM状态对于获得期望的结果是至关重要的。而且，LSTMs可能根本不需要简单的时间序列。

关键词:LSTM;RNN;异常检测;时间序列;深度学习

Abstrakt

为avvik -kelsedetektion i tidsseriedata提供长期短期内存(LSTM)。我的数据来自于我的数据，这是我的数据。这是一种非常有效的治疗方法。这是一种非常有效的方法，它可以帮助我们提高效率。在indata中，LSTM ar ocksa jamforda med vanliga neuronnat med fasta tidsfons-ter。在这个实验中，我们发现了一种新的方法，它可以使我们的研究成果得到广泛的应用。这是一种非常有效的方法。

关键词:LSTM;RNN;avvikelsedetektion;tidsserier;djupt larande

确认

我很感激Magnus Boman对他的时间、反馈和真诚的善意。他在斗争时期的指导是完成这篇论文的关键。我要感谢我的导师，Daniel Gillblad，感谢他对论文以及未来的投入和想法。我向蒙娜·马蒂和里克·科斯特致以最诚挚的问候，感谢他们给我机会在爱立信和他们的支持下做这篇论文。我还想感谢爱立信的整个机器智能团队，Kista，欢迎我加入这个团队，以及他们在项目中的参与。我要感谢我的对手，斯塔凡·阿尔德福克和安德烈·阿齐尼对我的工作的批评。特别感谢我的家人和朋友对我的持续鼓励和激励。最后，我要感谢我的妻子，Deepta，没有她的爱和支持，这一切都不可能实现。

策略!

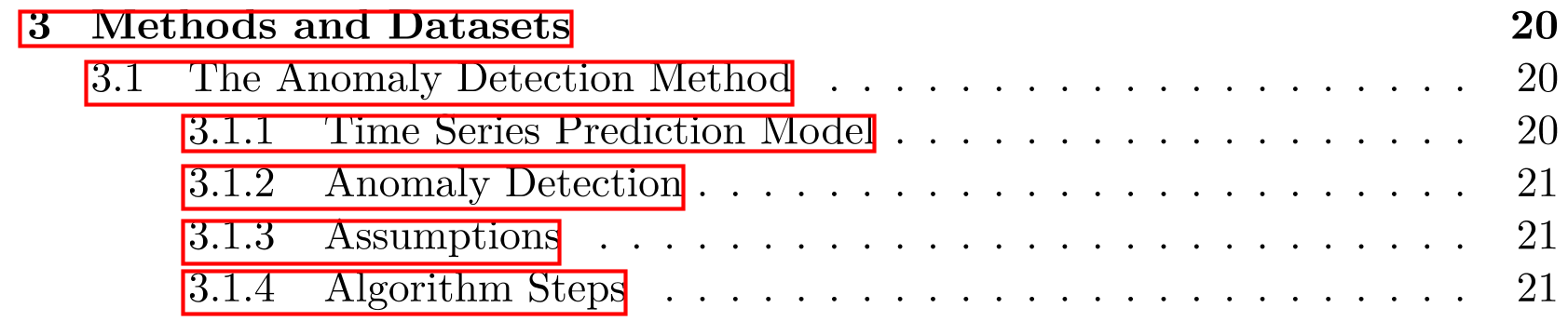
谢谢你！

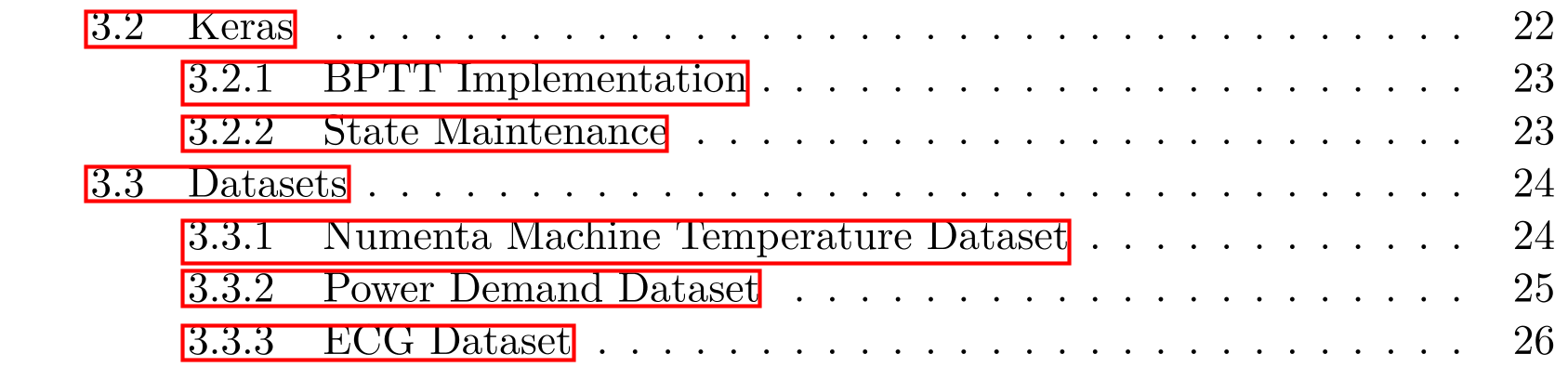
内容

缩写

|  |  |  |
| --- | --- | --- |
| 1 | 介绍  1.1异常检测。。。。。。。。。。。。。。。。。。。。。。。。。。。。1.2深学习。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。1.3问题和贡献。。。。。。。。。。。。。。。。。。。。。。。。1.4目的和目标。。。。。。。。。。。。。。。。。。。。。。。。。。。。1.5道德和可持续性。。。。。。。。。。。。。。。。。。。。。1.6方法。。。。。。。。。。。。。。。。。。。。。。。。。。。。  1.6.1项目环境。。。。。。。。。。。。。。。。。。。。  1.7界定。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。1.8概述。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。 | 1 1 2 3 4 4 5 5 6。 |

|  |  |  |
| --- | --- | --- |
| 2 | 相关的理论  2.1神经网络。。。。。。。。。。。。。。。。。。。。。。。。。。  2.1.1培训NNs。。。。。。。。。。。。。。。。。。。。。。。。。。。2.1.2深度学习和深层神经网络……  2. . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . .2.3 RNNs。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。  2.3.1培训RNNs。。。。。。。。。。。。。。。。。。。。。。。  2.4 LSTM。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。  2.4.1 LSTM忘了盖茨。。。。。。。。。。。。。。。。。。。。2.5深RNNs。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。2.6相关工作。。。。。。。。。。。。。。。。。。。。。。。。。。。。  2.6.1对时态数据的异常检测……2.6.2 RNNs用于异常检测…… | 7 7 9 10 11 12 13 14 15 16 17 17 18。 |





|  |  |  |
| --- | --- | --- |
| 4 | 实验和结果  4.1主要结果。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。  以下4.4.1数据预处理。。。。。。。。。。。。。。。。。。。。。。。4.1.2 Numenta Machine温度数据集……4.1.3电力需求数据集。。。。。。。。。。。。。。。。。。4.1.4心电图数据集。。。。。。。。。。。。。。。。。。。。。。。。  4.2维护LSTM状态。。。。。。。。。。。。。。。。。。。。。。。。3 .用固定时间窗的前馈NNs . . . .…4.4其他实验。。。。。。。。。。。。。。。。。。。。。。。。。  4.1.1 Lookback。。。。。。。。。。。。。。。。。。。。。4.4.2预测精度与异常检测…… | 28 2828 2830 334 37 39 39 40。 |

讨论42 5.1异常检测数据集、指标和评价……42

5.2正常的假设。。。。。。。。。。。。。。。。。。。。。。。43 5.3 LSTMs vs前馈NNs。。。。。。。。。。。。。。。。。。。。。。时间异常检测的LSTMs . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . .45 5.5 LSTMs:进化和未来……46

6结论47 6.1未来工作。。。。。。。。。。。。。。。。。。。。。。。。。47

参考书目48

缩写

AI人工智能

API的应用程序编程接口

BPTT反向传播通过时间

CEC常数误差旋转木马

款深层神经网络

DRNN深度递归神经网络。

格勒乌封闭的复发性单元

隐马尔可夫模型HMM的识别

HTM分层时间记忆

LSTM长期短期记忆

大中型企业最大似然估计

均方误差均方误差

神经网络的神经网络

PD概率密度

ReLU修正线性单元

RNN递归神经网络

SGD随机梯度下降法

第一章

介绍

在本论文中，我们探讨了LSTM RNNs在时间序列无监督异常检测中的应用。该项目是在瑞典爱立信实验室的机器智能研究小组内进行的。

1.1异常检测

异常检测是指在数据中发现偏离正常行为的实例或模式的问题。根据上下文和域，这些偏差可以称为异常、异常值[1]。在本文中，使用的术语是异常。异常检测应用于各种领域，如金融交易欺诈检测、工业系统故障检测、入侵检测、音乐流媒体服务中的人工bot侦听器识别等。异常检测很重要，因为异常常常表明有用的、关键的、可操作的信息，这些信息可以使企业和组织受益。

异常可分为四类[1]:

1。点异常:如果数据点与数据点的其余部分有很大的不同，则被认为是点异常。数据集的极端值。

在这一类。

2。集体异常:如果有一组相关的点是正常的，但如果综合起来是异常的，那么集合就是一个集合异常。

偏离正常模式的时间序列，属于集体异常。

3所示。上下文异常:如果一个数据点在特定的环境下被查看时是异常的，而正常情况下则被认为是上下文异常。

上下文通常以附加变量的形式出现，例如时间或空间属性。如果某些上下文属性存在，则点或集体异常可能是上下文异常。最常见的例子。

1

第1章。介绍

当一个点在正常范围内，但不符合预期的时间模式时，就会出现在时间序列数据中。

4所示。更改点:该类型是时间序列数据特有的，指的是典型模式变化或演化的时间点。变化点是不

总是被认为是异常。

这些类别对所采用的异常检测算法的类型有重要影响。

异常检测被认为是一个难题[1], [2]。异常被定义为偏离正常模式。然而，要提出一种规范的定义并不容易，它解释了正常模式的每一个变化。定义异常更加困难。异常是罕见的事件，并且不可能预先知道所有类型的异常。此外，异常的定义在不同的应用程序中也有所不同。虽然通常认为异常和正常点是由不同的过程产生的。

构建和评估异常检测系统的另一个主要障碍是缺乏标记的数据集。虽然异常检测一直是一个被广泛研究的问题，但在基准数据集[2]中仍然缺乏共识。在现实世界的许多应用程序中，异常都代表着关键的故障，这些故障代价太高，难以获得。在某些域中，有足够的容忍度，并且在公差间隔之外的任何值都可以标记为异常。虽然在许多情况下，标记异常是一个耗时的过程，但需要了解潜在物理过程的人类专家需要对异常进行注释。

对时间序列的异常检测提出了其独特的挑战。这主要是由于时间序列分析所固有的问题，被认为是数据挖掘研究中最具挑战性的十个问题[3]。事实上，时间序列预测与时间序列异常检测密切相关，因为异常是偏离期望值的点或序列[4]。

1.2深度学习

近年来，深度学习成为最受欢迎的机器学习技术之一，它为一系列的监督和非监督任务提供了最先进的结果。深度学习成功的主要原因是它能够学习与手边任务相关的高级表示。这些表示是自动从数据中学习的，很少或根本不需要人工特性工程和领域专门知识。对于序列和时间数据，LSTM RNNs已经成为了选择的深度学习模型，因为它们的能力。

2

第1章。介绍

远程学习模式。

由于采集标签存在问题，异常检测主要是一个无监督问题。虽然最近的重点和成功的深度学习研究都在监督学习中，但在未来几年，无监督学习预计将会得到更大的重视[5]。

1.3问题和贡献

正如上面所讨论的，异常检测有几个固有的挑战。这些与异常的定义有关，评估异常检测算法，以及数据集的可用性。在项目的过程中，我们面临着同样的问题，影响了我们对数据集的选择和评价方法的选择。我们将在后面的部分中讨论这些选择。另一个问题与对LSTM的理解有关。虽然LSTM已经成为顺序数据选择的机器学习模型，但是它的工作和局限性仍然是很不了解的。由于其复杂的体系结构，LSTM被看作是一个黑盒，不太清楚不同组件的角色和意义。

我们的工作有三个贡献。首先，我们解释了LSTMs的需求，并研究了LSTM体系结构，以说明为什么LSTMs适用于顺序和时间数据。其次，我们研究了LSTMs如何作为通用异常探测器来检测各种异常。第三，我们展示了不同的参数和架构选择如何影响LSTMs的性能。

1.4目的和目标

在无监督的异常检测中，只有很少的尝试使用LSTMs。在本论文中，我们尝试为时间序列建模和异常检测提供一个改进的LSTMs理解。其目的不是要开发一种高级的异常检测算法，而是要了解什么使LSTMs成为时间序列建模和异常检测的一个很好的选择，以及它们的局限性。

本文的目的是帮助爱立信了解LSTMs对异常检测的适用性。因此，本文的研究具有探索性。爱立信的机器智能研究小组一直在研究异常检测算法，目的是将它们纳入Erics-son的众多产品和服务中。该项目的结果将是一个时间序列异常检测算法，它的实现，并分析其在不同数据集上的性能。

3

第1章。介绍

1.5伦理和可持续发展

异常检测通常用于防范许多非法活动，例如金融交易中的欺诈预防/检测，以及安全系统中的入侵检测。由于我们使用的是公开可用的数据集，所以没有关于从人的主题收集数据的问题。使用公共数据集也可以减轻隐私问题。我们不主张优于其他作品，不伪造任何结果或数据，并采取适当的谨慎避免抄袭，并在需要时给予适当的引用。

测量异常检测技术对环境的直接和间接影响是值得的。

•一阶效应:异常检测系统是软件系统，对环境或生产的关注没有直接的影响，

废物、有害的副产品或污染。虽然异常探测系统需要计算机硬件资源来实现，但是这些系统的能耗和其他影响可以忽略不计。

•二阶效应:异常检测系统对实现它们的组织和各方有积极的影响。他们可以防止货币

在网络系统中发现财务欺诈和其他违法行为的损失。它们还可以通过检测工业机器的故障来减少浪费和提高生产率，从而允许及时采取纠正措施。

•本项目使用的异常检测算法是基于机器学习技术。关于如何增加adop的争论正在进行中。

机器学习和人工智能(AI)可能对社会有害。这些不良影响包括许多体力工作的丧失，甚至是“全知的邪恶AI”。我们确实承认，其中一些担忧并非完全没有根据，但对这种辩论的讨论超出了本报告的范围。就异常检测而言，由于任务的规模或复杂性，人工检查是不可能的。

1.6方法

在这个项目中，我们主要使用[6]所描述的定量研究方法。首先，对研究领域的主要挑战进行了文献研究，并利用问题领域的最新研究进展。采用实验研究方法，采用演绎法。该算法的实现是在探索性和迭代性的，从简单到更复杂的技术。该算法在各种数据集上进行评估，以确定其有效性。执行了严格的质量保证，以确保使用了软件库。

4

第1章。介绍

表1.1:项目中使用的软件库。

库版本Keras 2.0.3

TensorFlow 1.0.0

sickit-learn 0.18.2 GPyOpt 1.0.3

正确和代码没有受到任何可能影响实验结果的缺陷或缺陷。对结果进行了仔细的分析，以评价LSTMs的实用性。我们将讨论该方法的优缺点，并就其使用提出建议，并对今后的工作提出建议。

Python编程语言(2.7.12版本)用于为项目编写代码。所使用的主要软件库及其版本列在表1.1中。为了支持该研究的可重复性，在GitHub1上已经提供了用于该项目的整个代码库。

1.6.1项目环境

这项工作是在爱立信的机器智能研究小组内进行的，该小组位于瑞典的基斯塔。该集团专注于将机器学习和相关技术应用于与爱立信相关的问题。我加入了这个团队，参与了爱立信与瑞典学术界和业界合作的一个具体项目。该项目的重点是利用来自制造业的传感器数据进行异常检测和预测维护。

1.7的界定

在这个项目中，我们只关注使用LSTMs的无监督异常检测，并且不讨论其他替代技术。虽然多个软件包提供了LSTMs的实现，但我们使用了带有TensorFlow3后端的Keras2。Keras为支持快速实验的神经网络提供了一个高级API。Keras很容易使用，但是可以非常严格，并且构建定制的实现并不简单。这与LSTMs在项目中使用的方式有关，并在第3.2节中解释。另一种选择是直接使用10 - sorflow，它提供了比Keras更灵活、更低级的API。然而，TensorFlow的API最近发生了重大变化。我们发现

1 https:/ /github.com/akash13singh/lstm\_anomaly\_thesis 2 https:/ / keras.io /

3 https:/ /www.tensorflow.org/

5

第1章。介绍

文件没有更新，一些重要的细节也不见了。由于项目不需要定制的LSTM实现，所以我们决定使用Keras。其他选择的原因，包括数据集的选择、指标和异常检测方法都在相应的部分中提供。

1.8概述

报告的其余部分是这样组织的:在第2章中，我们概述了对理解项目其余部分所做的工作至关重要的理论和概念。本章还包括有关工作的一节。接下来，我们将在第三章中介绍该项目中使用的模型和方法。然后在第四章中给出我们的实验、结果和评价。第5章讨论了所使用的方法、实验和结果。最后，我们在第六章总结了该报告，并对今后的工作提出了一些设想。

6

第二章

相关的理论

2.1神经网络

神经网络(NN)是一种受生物大脑功能和结构启发的机器学习模型。一个NN由称为节点或神经元的简单计算单元组成。一个神经元接收输入边的输入，通过相应的边权值将输入相乘，然后应用一个叫做激活函数的非线性函数，来加权和产生一个输出。神经元的工作如图2.1所示，可以用向量方程2.1表示，其中x、w、b、f、y分别表示输入向量、权重向量、神经元偏差、元素-wise相乘、激活函数和神经元输出。

y(x) = f(w x + b) (2.1)

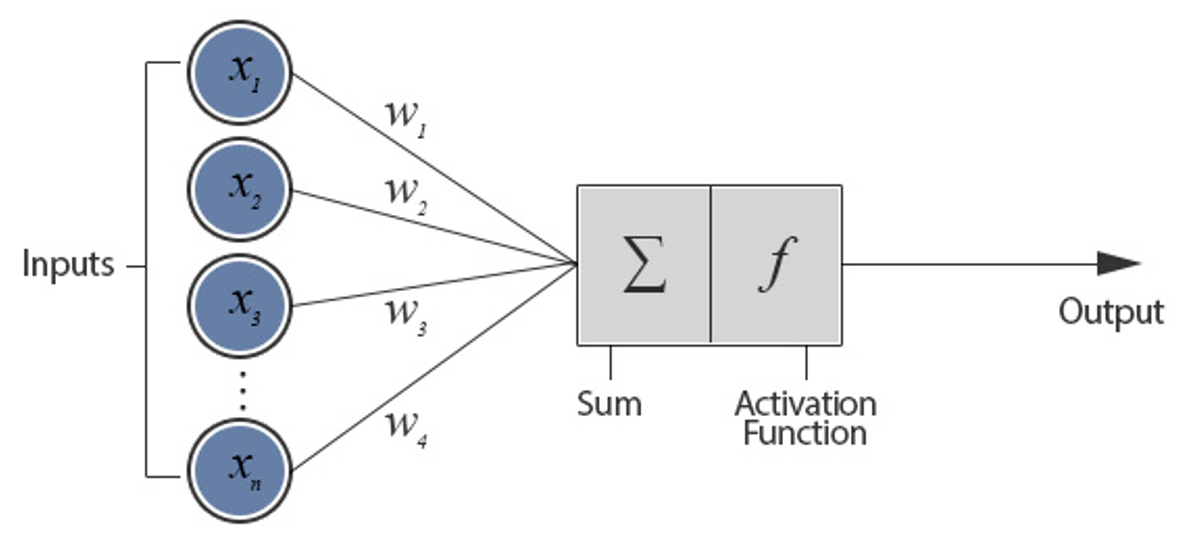


图2.1:神经元的功能。单个神经元的输出是其输入的加权和的非线性函数。非线性是由激活函数引入的。图像改编自[7]。

典型的激活功能包括:物流乙状结肠(σ),双曲正切,和修正线性单元(ReLU)[5]。函数由方程2.2、2.3和2.4定义。

7

第二章。相关的理论

分别。对于需要预测连续值的回归问题，使用线性激活。线性激活应用于等式2.5中所示的恒等函数。

σ(z)= 1

1 + e−z

(2.2)

双曲正切(z)= e

z

e−−z

ez + e−z

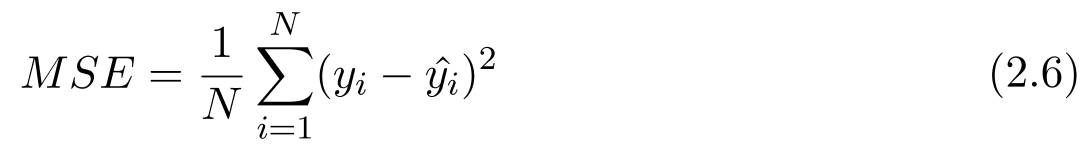
(2.3)

ReLU(z) = max(0, z) (2.4)

(z)= z(2.5)

基本的前馈网络如图2.2所示，由多个神经元组成，也称为单元，由层组织构成一个网络。每一层的神经元通过一组有向的边缘连接到上一层的所有神经元。每条边都有相应的权重。第一层接收输入，被称为输入层。最后一层称为输出层，它产生NN的输出。剩下的层被统称为隐藏层。由于信息流是从输入层到输出层，层结构中隐含了一个层次结构。输入层也被称为底层和输出层作为顶层。每个神经元的输出都是从底层向上移动，直到网络的输出在顶层生成。

前面描述的前馈神经网络用于监督学习任务[5]。在训练过程中，网络会随着输出一起呈现输入数据。为了便于学习，建立了测量网络输出与期望输出之间距离的损失函数。通常用于回归问题的损失函数(即预测一个连续值)ismean平方误差(MSE)。MSE计算如公式2.6所示,其中N是观测的数量,易建联表示真正的价值,用ŷ我的预测价值。MSE度量了预测值和真实值之间的平均平方距离。真实值与预测值之间的差值也称为误差或残差。



8

第二章。相关的理论

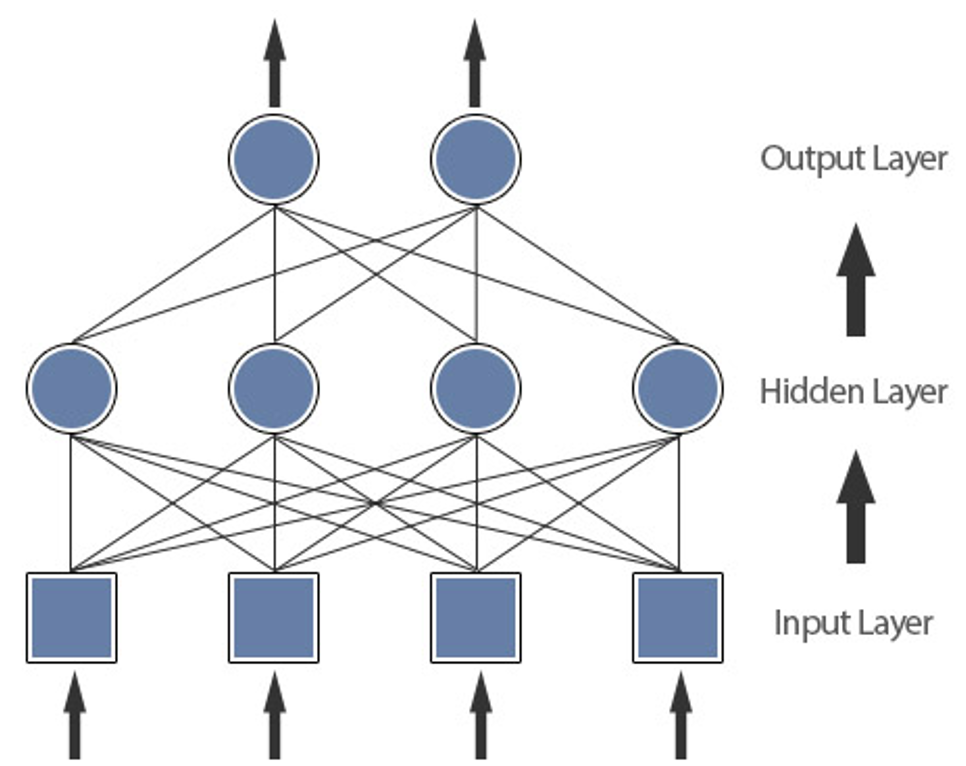


图2.2:一个有一个隐藏层的前馈NN。神经元由圆表示。一层的每个神经元都连接到上一层的所有神经元。在没有任何处理的情况下，输入层节点不是技术上的神经元。图像改编自[7]。

2.1.1训练得到

将学习问题转化为优化(误差最小化)运动，通过调整NN的参数使损失函数最小化。用于训练NNs的优化算法称为梯度下降法。梯度下降包括计算损失函数与网络参数(权重和偏差)的梯度。用来计算梯度的方法称为反向传播[8]，它基于导数的链式法则。梯度是对网络参数中一个小变化对应的损失值变化的度量。标量值称为学习速率(γ)是用于更新的参数(θ)相反方向的梯度,根据方程2.7。通过对训练数据进行多次传递，迭代过程是迭代完成的。通过训练数据的传递被称为一个时代，在每一个时代之后，这些参数会接近它们的最优值，从而使损失函数最小化。

θ=θ−γ∂L(θ)∂θ(2.7)

如果数据集大小很大，那么计算整个数据集的损失和梯度可能太慢，在计算上不可行。因此，在实践中，通常使用一种称为随机梯度下降(SGD)的梯度下降法。在SGD数据中，将数据划分为被称为批次的子集合，在计算损失函数超过一个批次后，对参数进行更新。其他流行的变体有:rmc - sprop, AdaGrad, Adam[9]。在其中的一些变体中，还有一个额外的参数。

9

第二章。相关的理论

当参数接近最优值时，被称为衰变被用来逐渐降低学习速率。

在训练NNs中经常遇到的问题是过度拟合。当模型试图适应训练数据中的噪声时，就会发生过度拟合，通常是使用比所要求的更复杂的模型的结果。在过度拟合的情况下，模型在训练数据上表现良好，但在新数据上表现不佳。有几种方法可以防止过度拟合。在早期停止的一个小子集训练数据作为验证集。在每一个时代的价值损失函数在训练集相比,价值损失验证验证设置。如果开始增加尽管训练集上的损失减少,这是一个过度拟合的迹象,模型训练可以停止了。在深度学习中常用的另一种方法是辍学。在辍学率中，在每个训练阶段中随机抽取一定百分比的NN连接。

需要注意的是，网络参数(权重和偏差)是通过训练算法来学习的。另一方面，学习速率、辍学率、训练批量大小、衰减等参数是学习算法的参数，需要由用户设置相应的值。后一个参数统称为超参数。对于NNs、梯度下降和反向传播的详细研究，可以参考[10]的第5、6、7和8章。

2.1.2深度学习和深层神经网络。

传统机器学习管道的一个重要组成部分是特征工程[11]。传统的机器学习算法需要精心设计的特性，并且不能很好地处理原始数据。然而，特性工程并不简单，需要相当多的领域专门知识。深度学习模型成功的主要原因之一是能够自动学习与手头任务相关的高级表示。深度神经网络(DNNs)是一种多隐藏层叠加在一起的神经网络。每一层都是一个非线性模块，它接收其上一层的输出。越来越多的复合/抽象的特性从下到顶层学习。因此，DNN类似于一个处理管道，每个层都执行部分任务，并将其输出交给下一层。

深度学习技术在从计算机视觉到语言翻译的各个领域都取得了最先进的成果[10]。许多不同的因素促成了这一成功:大型标记数据集的可用性、计算机工程、分布式系统以及包括gpu在内的计算能力的进步。

10

第二章。相关的理论

2.2序列数据需要RNNs。

在研究RNNs之前，我们有必要了解为什么需要RNNs和NNs在建模顺序数据方面的缺点。

NNs的一个主要假设，事实上许多其他的机器学习模型是数据样本的独立性。然而，这种假设并不适用于自然连续的数据。语言、语言、时间序列、视频等都表现出个体之间的相互依赖。NNs单独处理每个数据样本，从而失去利用这一序列信息获得的好处。一种解释顺序依赖的机制是将固定数量的连续数据样本连接在一起，并将它们视为一个数据点，类似于在数据流上移动一个固定大小的滑动窗口。该方法在[12]中使用NNs进行时间序列预测，[13]用于声学建模。但是正如[12]中提到的，这种方法的成功依赖于找到最佳窗口大小:小窗口大小不能捕获较长的依赖关系，而较大的窗口大小则会增加不必要的噪声。更重要的是，如果数据中有超过数百个时间步骤的远程依赖，那么基于窗口的方法就不会扩展。传统NNs的另一个缺点是不能处理可变长度序列。对于许多领域，如语音建模，语言翻译的输入序列长度不一。

隐马尔可夫模型(HMM)[14]可以在不需要固定大小窗口的情况下对序列数据进行建模。HMMs将观察到的序列映射到一组隐藏状态，通过定义隐藏状态之间的转换概率分布，以及观察值与隐藏状态之间的关系。HMMs基于Markov属性，根据该属性，每个状态仅依赖于前一个状态。这严重限制了HMMs捕获远程依赖性的能力。此外，HMMs的空间复杂度随着状态的增加而增加，并且没有很好地扩展。

RNNs在一个时间内处理一个元素的输入序列，并保持一个隐藏的状态向量，作为过去信息的记忆体。他们学会有选择地保留相关信息，从而允许他们跨几个时间步骤捕获依赖项。这使得他们可以利用当前的输入和过去的信息，同时进行未来的预测。所有这些都是由模型自动学习的，不需要了解数据中的周期或时间依赖性。RNNs消除了固定大小时间窗口的需要，也可以处理可变长度序列。此外，可以由NN表示的状态数在节点数中是指数级的。

11

第二章。相关的理论

2.3 RNNs

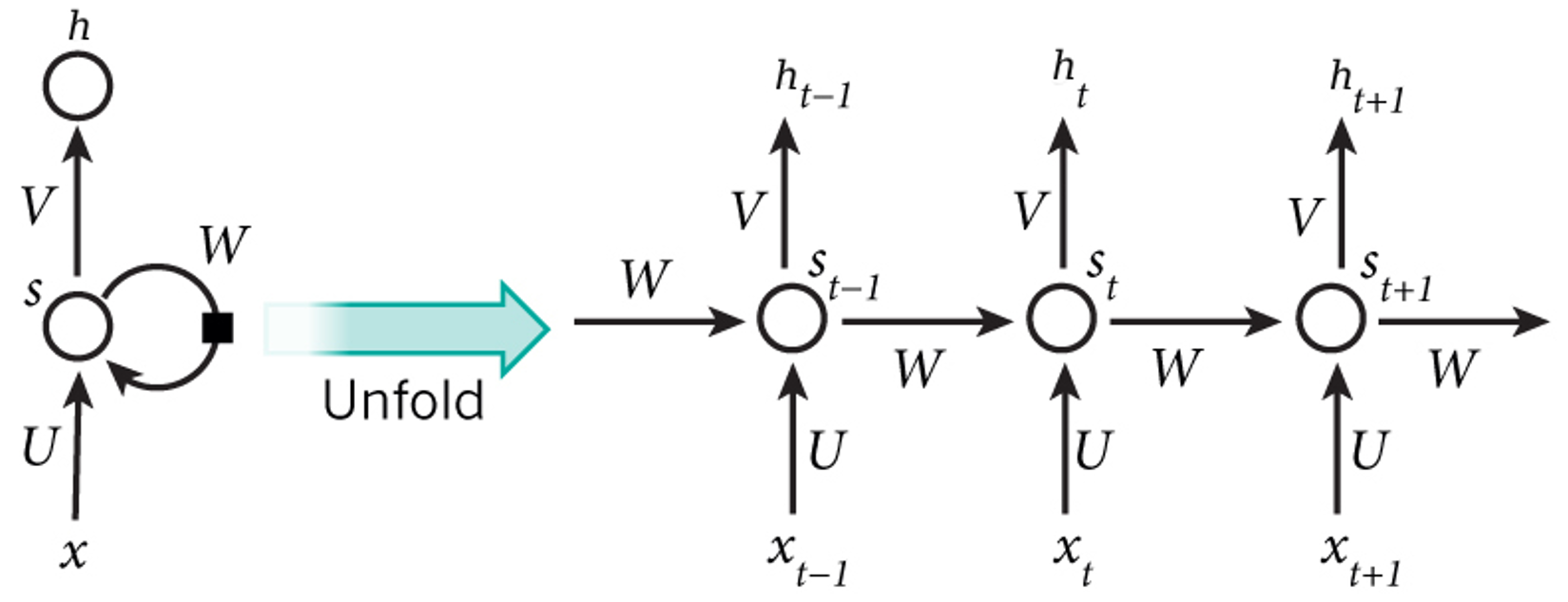


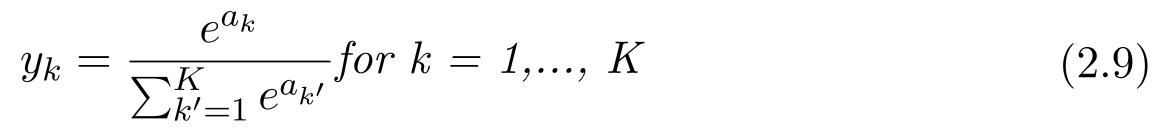
图2.3:标准RNN。图的左边是一个标准的RNN。隐藏单元中的状态向量是由s表示的，在右侧是相同的网络，在时间内展开描述状态是如何随着时间而建立的。图像改编自[5]。

RNN是一种特殊类型的NN，适合处理顺序数据。RNN的主要特征是一个状态向量(在隐藏单元中)，它保留了序列中所有先前元素的内存。最简单的RNN如图2.3所示。正如可以看到的，RNN有一个反馈连接，它连接了隐藏的神经元。在时间t,RNN接收作为输入当前xt和隐藏的状态序列元素从之前的时间步圣−1。然后将隐藏状态更新为st，最后计算网络ht的输出。在这种情况下，当前的输出ht依赖于所有以前的输入(对于t t)。U是在输入和隐藏层之间的权重矩阵，与传统的NN类似。W是在一个隐藏状态到下一个隐藏状态之间的周期性过渡的权重矩阵。V是隐藏到输出转换的权重矩阵。公式2.8总结了每一步所进行的所有计算。

st = (Uxt +Wst 1 + bs)

软max(V st + bh) (2.8)

2.8的softmax表示在多类分类问题中作为输出层的激活函数的softmax函数。softmax函数确保所有输出范围从0到1，并且它们的和为1。公式2.9指定了K类问题的softmax。



12

第二章。相关的理论

如果一个标准RNN在操作过程中考虑它的行为，那么它本身就是一个深NN。如图右侧所示，一旦网络及时展开，就可以将其视为一个深度网络，其层数与输入序列中的时间步数相等。由于每个时间步都使用相同的权重，RNN可以处理可变长度序列。在每次输入新输入时，由于隐藏状态st更新的方式(公式2.8)，信息可以在RNN中以任意数量的时间步骤流，允许RNN保持对所有过去信息的记忆。

2.3.1培训RNNs

RNN培训是通过展开RNN并为每个时间步骤创建一个模型的拷贝来实现的。在图2.3的右侧展开的RNN可以被视为一个多层NN，并且可以通过类似于反向传播的方式进行训练。这种训练RNNs的方法称为通过时间反向传播(BPTT)[15]。

理想情况下，RNNs可以通过BPTT训练来学习任意长序列的远程依赖关系。训练算法应该能够学习和调整权重，将正确的信息放入内存中。在实践中训练RNNs是困难的。事实上，标准RNNs即使在输出和相关输入被间隔为10个时间步骤时，也会表现得很差。现在众所周知，标准RNNs不能通过长时间间隔学习依赖关系[16]，[17]。使用BPTT训练RNN需要在多个时间步骤中反向传播错误梯度。如果我们考虑标准的RNN(图2.3)，在每一步中，循环边的权重是相同的。因此，反向传播错误包括将错误梯度与相同的值相乘，然后除以增益。这导致梯度要么太大，要么衰减到零。这些问题分别称为爆炸梯度和消失梯度。在这种情况下，模型学习根本不收敛，也可能会花费大量的时间。确切的问题取决于反复的边缘重量的大小和使用的特定的激活函数。如果权重的大小小于1，并且使用sigmoid激活(方程2.2)，消失的梯度更有可能，反之，如果震级大于1，而ReLU激活(方程2.4)则使用爆炸梯度更有可能[18]。

针对训练RNNs中长期依赖的问题，提出了几种解决方法。这些内容包括对培训过程的修改以及新的RNN架构。在[18]中，如果梯度的规范越过一个预定义的阈值，则建议将梯度缩小。这种被称为梯度剪裁的策略已经被证明能够有效地减轻爆炸的梯度问题。为了解决渐变问题，[18]引入了一个类似于L1的惩罚项，L2正则化惩罚用于防止在NNs中过度拟合。然而，正如所指出的，使用约束来避免消失的梯度，使得爆炸梯度更有可能。引入了LSTM体系结构。

13

第二章。相关的理论

[19]为了对抗消失的梯度问题。与标准RNNs相比，LSTM网络在学习长期依赖性方面非常有用，并且已经成为RNN最流行的变体。

2.4 LSTM

LSTM可以学习任意长时间间隔的依赖关系。LSTM通过一个称为LSTM单元或块的复杂架构替换一个普通的神经元，克服了消失的梯度问题。LSTM单元是由以特定方式连接的简单节点组成的。[19]中引入的LSTM体系结构的主要组件是:

1。恒定误差旋转(CEC):一个与单位重量有经常性联系的中央单位。循环连接表示一个反馈循环。

时间步长等于1。CEC的激活是作为过去信息记忆的内部状态。

2。输入门:保护存储在CEC中的信息不受无关输入干扰的乘法单元。

3所示。输出门:用CEC中存储的信息保护其他单位不受干扰的乘法单元。

输入和输出门控制进入CEC。在训练过程中，输入门学习何时在CEC内释放新的信息。只要输入门的值为零，就不允许内部信息。类似地，输出门学习什么时候让信息从CEC中流动。当两个门都关闭(激活0)信息或激活被困在记忆细胞内。这使得错误信号可以跨多个时间步骤(借助于单位重量的循环边)，而不会遇到渐变的问题。在2.3.1节中讨论的梯度剪切问题，解决了爆炸梯度的问题。

上面描述的标准LSTM在学习远程依赖性方面比RNNs更好。然而，在[20]中发现了一个缺点。在长时间连续的输入流中，没有显式标记的序列起始点和端点，LSTM状态将无限增长，最终导致网络变得不稳定。该模型将无法学习数据中的循环或序列。除非输入流被手动地分割为适当大小的序列，否则LSTM状态将不会被重置。理想情况下，LSTM应该学会在处理完一个序列之后重新设置内存单元的内容，然后再开始一个新的序列。为了解决这一问题，在[20]中引入了一个新的LSTM体系结构，并引入了遗忘门。当启动新的序列时，忘记盖茨学习重置LSTM内存。但是，已经提出了许多其他的修改和LSTM体系结构的变体，这些都是在[21]中记录的。

14

第二章。相关的理论

变异有类似的表现。由于更简单的体系结构更可取，所以在本文中我们使用的是LSTM，而忘记了gates。接下来我们将更详细地描述这个体系结构。

2.4.1 LSTM与遗忘门。

一个带有遗忘门的LSTM单元的体系结构如图2.4所示，是本报告其余部分使用的体系结构。LSTM单元的主要组成部分是:

1。输入:LSTM单元接收由xt表示的当前输入向量，以及前一个时间步(通过循环边)表示的输出。

ht−1。对加权输入进行求和，并通过tanh激活，得到zt。

2。输入门:输入门读取xt和ht−1,计算加权和,适用于乙状结肠激活。结果它与zt相乘。

提供流入内存单元的输入。

3所示。忘记门:遗忘门是一种机制，通过它，LSTM可以在它们变老时重新设置内存内容，不再是rele。

vant。当网络开始处理一个新的序列时，这可能会发生。忘记门读xt和ht−1和乙状结肠激活适用于加权输入。其结果是，ft在前一阶段乘以细胞状态，即st 1，它允许忘记不再需要的内存内容。

4所示。记忆细胞:由CEC组成，具有单位重量的循环边。当前的细胞状态st是通过遗忘无关信息来计算的。

(如果有的话)从上一阶段开始，并接受当前输入的相关信息(如果有的话)。

5。输出门:输出门的加权和xt和ht−1和乙状结肠激活适用于控制信息将LSTM流出

单位。

6。输出:LSTM单元的输出，ht，通过将单元状态st通过tanh来计算，并将其与输出门相乘，ot。

LSTM单元的功能可以用下面的一组来表示。

15

第二章。相关的理论

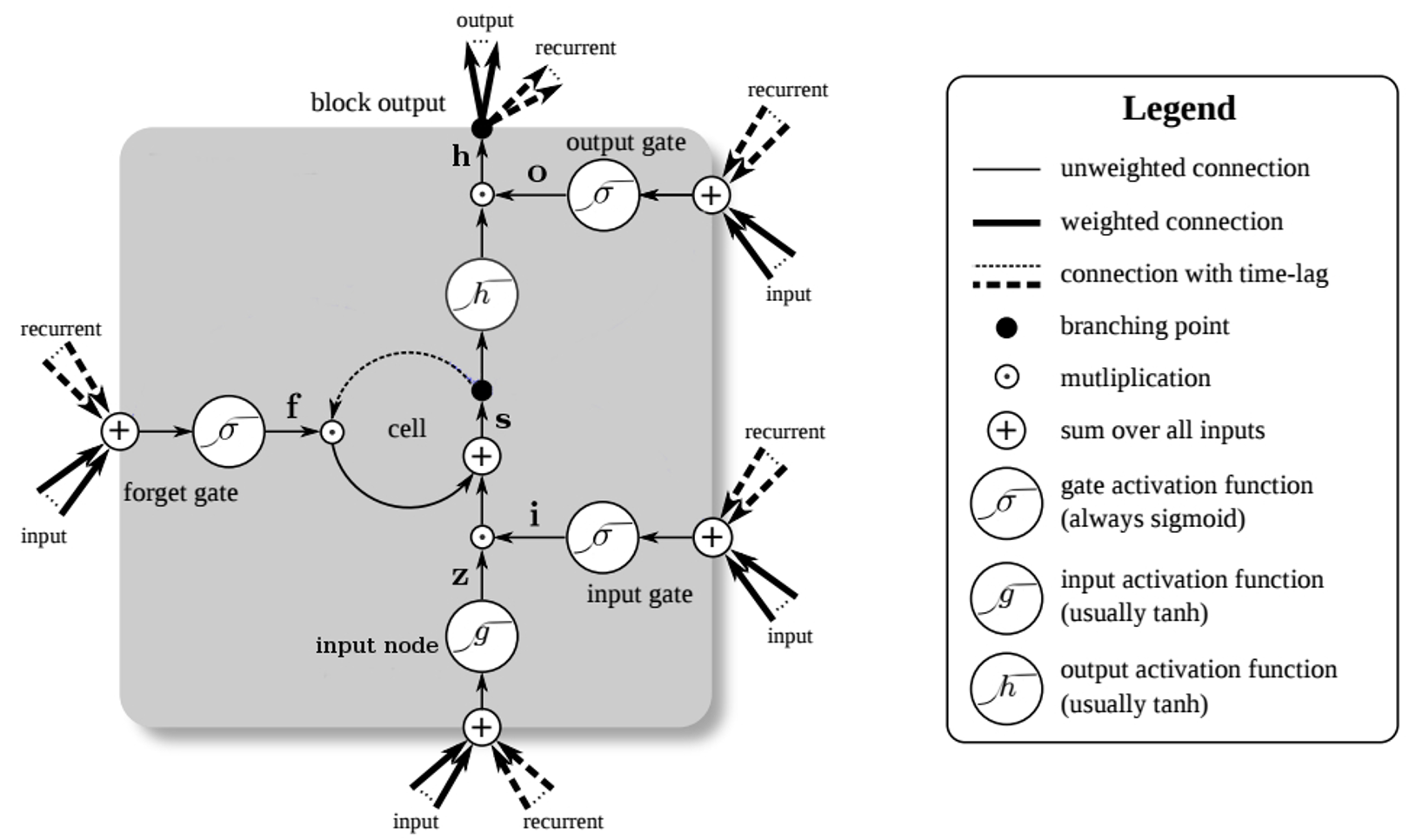


图2.4:带有遗忘门的LSTM单元。在[20]中引入了遗忘门的LSTM单元示意图。图像改编自[21]。

方程:

zt型=双曲正切(W zxt + Rzht−1 + bz)它=σ(W ixt + Riht−1 + bi)

ft = (W fxt +Rfht 1 + bf) = (W oxt +Roht 1 + bo) st = zt it + st 1 ft。

ht =双曲正切(st)不能

(输入)

(输入门)

(忘记门)(输出门)

(细胞状态)

(2.10)

(输出)

W∗年代输入重量、R∗s是经常性的重量,和b∗s是偏见。注:从这里开始，我们将使用RNN和LSTM RNN交换条件。

可以用LSTM单位表示RNN。在2.3节中提出的香草RNN架构将被称为标准RNN。还要注意，我们总是使用带有单个内存单元的LSTM单元。

2.5深RNNs

正如第2.1.2节所提到的，深度学习模型的成功是由于能够学习一种简单到复杂(抽象)特征的层次结构，这是由多层面的叠加所促成的。一个RNN可以被认为是一个DNN，当它在时间内展开时，每一步都有一个层，每个层都有相同的功能。

16

第二章。相关的理论

然而，在RNN中深度的目的与DNN不同。DNN在底层进行输入，在生成输出之前通过多个隐藏的非线性层处理信息。另一方面，RNN接受输入并在每次步骤中产生输出，在输入和输出之间只有一个非线性层。因此，在RNN深度中，仅用于维护旧信息的内存，但不像DNN那样提供信息的分层处理。RNN在两种情况下都是短的[22]:首先，在复杂的序列数据的情况下，需要通过许多非线性层进行层次信息处理;第二，当顺序数据如语音或时间序列包含需要在不同的时间尺度处理的模式时，但是RNNs在一个时间尺度上运行。

为了解决这些缺点，RNNs具有多个隐藏层或深RNNs (DRNN)，用于语音识别[23]和声学建模[24]。DRNNs也被称为堆叠RNNs，以表明多个RNN层被叠加在一起。我们可以互换使用这两个词。在一个DRNN中，每个层都可以有多个LSTM单元，一个层的输出序列作为下一层的输入序列。每一层的隐藏状态按公式2.11[23]计算。

hn t = H(W hn−1 hnhn−1 t + Rhnhnhn t−1 + bn H)(2.11)

其中H为方程2.10所给出的LSTM函数;n，从1到n，表示网络的第n层;t表示时间步长。W表示两层之间的前馈权值，R表示从一个时间步到下一个相同层的重复权值。网络输入被定义为h0 = x，由yt的网络输出计算为:

yhN t + by (2.12)

从公式2.11和2.12可以得到一些关于DRNN提供不同时间尺度的见解。第一层将构建输入信号的内存。下一层将会对第一层的隐藏状态进行记忆，从而有一个“更深入”的记忆，同时也经历了一个额外的非线性计算。接下来的每一层[22]。

2.6相关工作

2.6.1时间数据异常检测。

时间数据中的异常是上下文异常，提供上下文的时间。以办公室日常用电需求为例:

17

第二章。相关的理论

工作日和周末的需求都很高。然而，周末的高需求或工作日的低需求可能意味着一些不寻常的事情。因此，正常范围内的高(低)值在本周内变得异常。根据领域和用例，人们可能对点异常或集体异常感兴趣。另一个重要的考虑因素是数据的维度。如果数据集是多维的，具有代表不同时间序列的所有特性，那么可以使用多变量时间序列的方法。然而，同样常见的方法是，使用不考虑数据的时间方面的方法，并处理多维空间中的发现点异常。在本文中，我们只考虑单变量数据集，但所开发的模型可以很容易地推广到多变量时间序列。

时间异常检测的一种方法是建立预测模型，并利用预测误差(预测值与实际值之间的差值)来计算异常值[4]。采用了多种简单到复杂的预测模型。在[25]中，一个简单的基于窗口的方法被用来计算最近值的中值作为预测值，而预测错误的阈值被用来标记离群值。在[26]中，作者建立了一个超前的预测模型。如果数据点超出了预测误差的标准偏差值，则被认为是异常值。作者比较了不同的预测模型:朴素预测器、最近的集群、多层感知器和单层线性网络。

在[27]中提出了一种网络新颖性检测的框架，该框架计算每个识别出的新事物的置信分数。作者还开发了一种具体的算法，利用支持向量回归模型对时态数据进行建模。在[28]中使用了一个多变量的ARIMA模型，只在正常数据(没有任何异常情况下)进行训练。分段线性模型用于模型时间序列[29]。为了比较新的时间序列和参考序列，计算了相似度度量。然后，通过将相似度度量与使用参考序列的标准偏差的阈值集进行比较，突出显示不同寻常的模式。[30]介绍了天然气消费时间序列异常检测的概率方法。然而，预测方法预测使用其他自变量的消费水平，并且不使用数据的时间方面。所采用的预测模型为线性回归，利用已知异常训练的贝叶斯极大似然分类器对新数据中的异常进行分类。

2.6.2异常检测RNNs。

由于我们使用RNNs作为预测模型，我们在本节中回顾了利用RNNs进行时间异常检测的研究。在[31]的时间序列中，利用堆叠LSTM RNNs进行异常检测。该模型只需要一个时间步骤作为输入，并在整个输入序列中维护LSTM状态。模型训练

18

第二章。相关的理论

在正常数据上，并预测多个时间步。因此，每一个观察都有多个在过去不同时间做出的预测。多重预测被用来计算误差向量，这些向量是用多变量高斯分布来模拟的，以给出异常的可能性。该模型在4个真实的数据集上进行了测试，其中包括功率需求数据集，它的精度和召回率分别为0.94和0.17。同样的方法也用于[32]检测心电图数据集中的异常。根据作者的研究结果，他们发现LSTMs是一种可行的ECG信号异常检测方法。

在[33]LSTMs中，用于检测网络安全域中的集体异常。采用单重复层的LSTM RNN作为预测模型。大于设定值的预测误差表示点异常。保持一个循环数组来存储最近的预测错误。基于圆形阵列中的信息，计算了两个指标:一是异常的观测值的百分比;二是阵列中预测误差之和。如果两者都高于特定的阈值，则这些观测所对应的序列被标记为集体异常。该模型用于检测KDD 1999数据集中的异常。作者报告了0.86的召回，没有误报。他们还指出，可以实现1.0的召回，但代价是产生大量的误报。

上述所有工作都使用RNNs来模拟正常的时间序列模式，并以无监督的方式进行异常检测，仅使用标签来设置预测错误的阈值。RNNs还被用于时间序列分类模型的监督异常检测。这是一种可行的方法，除非有足够的标记异常。LSTM RNNs用于[34]中的时间序列分类，在边界网关协议数据中发现异常。作者报告说，使用固定大小的输入窗口和向下采样时间序列会提高性能。另一种有趣的方法是将RNNs与autoencoders结合，以模拟正常的时间序列行为。结果模型重构了时间序列序列，并利用重构误差进行异常检测。该方法在[35]中应用于[0]，用于声学新颖检测，[36]用于多传感器异常检测。

1 http:/ /kdd.ics.uci.edu/databases/kddcup99/kddcup99.html

19

第三章

方法和数据集

3.1异常检测方法。

项目中使用的异常检测算法主要有两个步骤。首先，建立一个总结预测模型，学习正常的时间序列模式，预测未来的时间序列。然后通过计算异常值对预测误差进行异常检测。

3.1.1时间序列预测模型。

我们使用LSTM RNN作为时间序列预测模型。模型以输入最近的p值和输出q的未来值作为输入。我们分别将参数p, q作为向后看和展望。网络由隐藏的重复层/层和输出层组成。每个数据集的隐藏的重复层数和每个层的单元数各不相同。连续两个连续层是完全连通的。为了避免过度拟合，在两个连续的层之间使用。输出层是一个完全连通的稠密NN层。输出层中的神经元数量等于前面的值，每个未来值都预测一个神经元。由于该模型用于回归，我们在输出层使用线性激活，MSE作为损失函数。该预测模型仅对正常数据进行训练，不存在任何异常现象，使其能够学习时间序列的正常行为。

预测多个时间步骤:我们尝试预测未来的多个时间步(类似于[31])。随着时间t的提前，模型预测了时间序列的下一个q值，即t+1, t+2…,t + q。预测多个时间步骤有两个目的。首先，将LSTM的能力作为时间序列建模器来展示，预测多个未来值是一个比较困难的问题，而不是一步一步的预测。第二，预测多个时间步骤提供了未来行为的早期概念。甚至有可能得到一个异常的早期迹象。考虑一个时长为5分钟的时间序列。提前6个步骤可以让我们了解时间的行为。

20.

第三章。方法和数据集

接下来的30分钟。如果有一些不寻常的东西，例如一个极端值，可以发出早期警报。然而，只有当真正的输入值变为可用时，异常检测才会发生。预测多个时间步骤的代价是预测的准确性。在我们的实验中，只有在预测精度仍然可以接受的情况下，我们才使用大于1的预测。使用的实际lookahead值是任意选择的。

3.1.2异常检测

将预测误差作为异常指标进行异常检测。预测误差的区别是预测在时间t−1和接收到的输入值在时间t。训练数据的预测错误使用高斯分布建模。利用最大似然估计方法计算了高斯、均值和方差的参数。在新数据中，计算错误的对数概率密度(PDs)作为异常值，其值越低，表明观测结果的可能性越大。一个包含正常数据和异常的验证集被用来设置一个日志PD值的阈值，该阈值可以将异常与正常的观测分开，并尽可能少地产生假阳性。一个单独的测试集用于评估模型。

3.1.3假设

我们的主要假设是，一个训练于正常数据的预测模型应该学习正常的时间序列模式。当该模型用于预测新数据时，与正常区域相比，该模型对异常区域的预测误差应该更高。这将使我们能够将错误的日志PD值作为异常值，并设置一个阈值，将异常与正常数据点分开。另一个假设是预测误差遵循高斯分布。

3.1.4算法步骤

LSTM RNN只对正常数据进行训练，学习正常的时间序列模式，并对预测精度进行优化。为此，每个数据集被划分为4个子集:一个训练集，N，只有正常的值;验证集，VN，只有正常值;第二个验证集VA，具有正常的值和异常;一个测试集，T，有正常的值和异常值。该算法的收益如下:

1。集合N用于训练预测模型。我们使用贝叶斯优化[37]来寻找超参数的最佳值:

学习速率和网络体系结构(每个层的隐藏层数和单位数)。如果预测accu-，我们将使用超过1的预测。

21

第三章。方法和数据集

生动的仍然是合理的。如果不需要预测多个时间步骤，并且需要最佳的预测精度，则可以将lookahead设置为1。

2。VN用于早期停止，以防止模型过度拟合训练数据。

3所示。利用高斯分布对N的预测误差进行了建模。用MLE估计分布的均值和方差。

4所示。将训练后的预测模型应用于VA，利用上一步计算的分布参数来计算错误的log PDs。

从VA开始，在日志PD值上设置一个阈值，该值可以分离异常，并尽可能少的假警报。

5。使用测试集T的预测错误来评估设置的阈值。

实际的实验需要修改。根据上面的步骤1，我们对模型进行训练，并对集合N的预测精度进行优化。根据我们的假设，模型学习了正常的时间序列行为，并且在集合VA和T上应该有更高的预测误差，从而允许我们做异常的脱位。然而，正如在4.4.2节中所讨论的那样，最适合预测的参数并不总是能给异常检测带来好的结果。在这种情况下，我们将优化阶段的参数作为起始点。通过重复步骤1至4，直到我们能够检测到set VA中的所有异常，该模型进一步手动调整。

贝叶斯优化是利用库GPyOpt1进行的。优化过程要求我们为每个超参数提供候选值。在文献中给出了学习率、辍学率等参数。对于网络体系结构，我们提供了任意选择的三个变量。

3.2 Keras

我们使用Python编程语言和Koras来编码和实现项目的实验。Keras是一个开源项目，它提供一个高级API来实现DNNs，并在其他深度学习库(如TensorFlow和Theano2)上运行。我们用Keras在TensorFlow上。Keras被选为它是为快速原型和实验设计的一个简单的API。它允许通过组合不同的层、激活函数、损失函数和优化器等模块化方式配置NNs。Keras提供了大多数标准深度学习构建块的框解决方案。但是，如果有人想要构建一个定制的或新的实现，Keras API可能非常有限，

1 https:/ / sheffieldml.github.io / GPyOpt /

2 http:/ /deeplearning.net/software/theano/

22

第三章。方法和数据集

而像TensorFlow这样的图书馆将是一个更好的选择。Keras包含了在[20]中描述的遗忘门的LSTM的实现。下面解释了两个重要的细节，这对于理解如何在Keras函数中实现LSTMs至关重要。

3.2.1 BPTT实现

在Keras中实现了BPTT的修改版本。在整个输入序列中展开RNN，由数百个甚至数千个时间步骤组成，计算效率不高。因此，在Keras RNN中，展开了最多的时间步骤。该参数通过输入机制提供。输入数据以三维形状数组的形式输入:(batch\_size, lookback, input\_dimension)。第二个参数lookback指定了RNN展开的时间步骤的数量。输入数据被分成重叠的序列，时间间隔为1。每一个序列都有相应的连续时间步长，并形成一个训练样本到RNN模型。在培训期间，BPTT只对单个样本进行回顾，以查看时间步骤。

3.2.2状态维护

Keras提供了两种不同的维护LSTM状态的方法。

1。默认模式:批处理中的每个样本都被假定为独立的，并且状态只在回显数字的单个输入序列上保持。

的时间步骤。

2。状态模式:在这个选项中，在不同的训练批次中保持单元状态。将当前批次的第i个样品的最终状态用作。

下一批的第i个样品的初始状态。在批处理中，单个样本仍然是独立的。使用有状态模式的一个常见错误是对训练样本进行洗牌。然而，为了在各个批次之间保持状态，假设在连续批次的样本之间进行一对一的映射，并且要小心不要对样本进行随机抽样。

单个样本之间的独立性看起来很奇怪。这种实现的动机来自于语言建模和语音识别任务，这些任务是驱动LSTM开发和实现的关键领域。在许多语言建模任务中，训练样本是单个的句子，而一个短的回视值等于最大的句子长度(用文字表示)就足以捕获必要的顺序依赖关系。因此，不同的样品可以单独处理。但是，对于许多域和数据集，这种行为可能是非常严格的。

23

第三章。方法和数据集

3.3数据集

如文献[2]所述，异常检测研究的一个主要问题是缺乏标记的基准数据集。许多发表的作品要么使用应用程序特定的数据集，要么生成合成数据集[38]。然而，这两种方法都有其缺陷。对于应用程序特定的数据集，很难判断异常检测算法对不同数据集的泛化程度。在合成数据集的情况下，该算法的异常和性能没有实际的有效性。为了防范这些问题，我们选择了来自不同领域的真实数据集。这些数据集已经被用于之前的异常检测工作。

3.3.1 Numenta机器温度数据集。

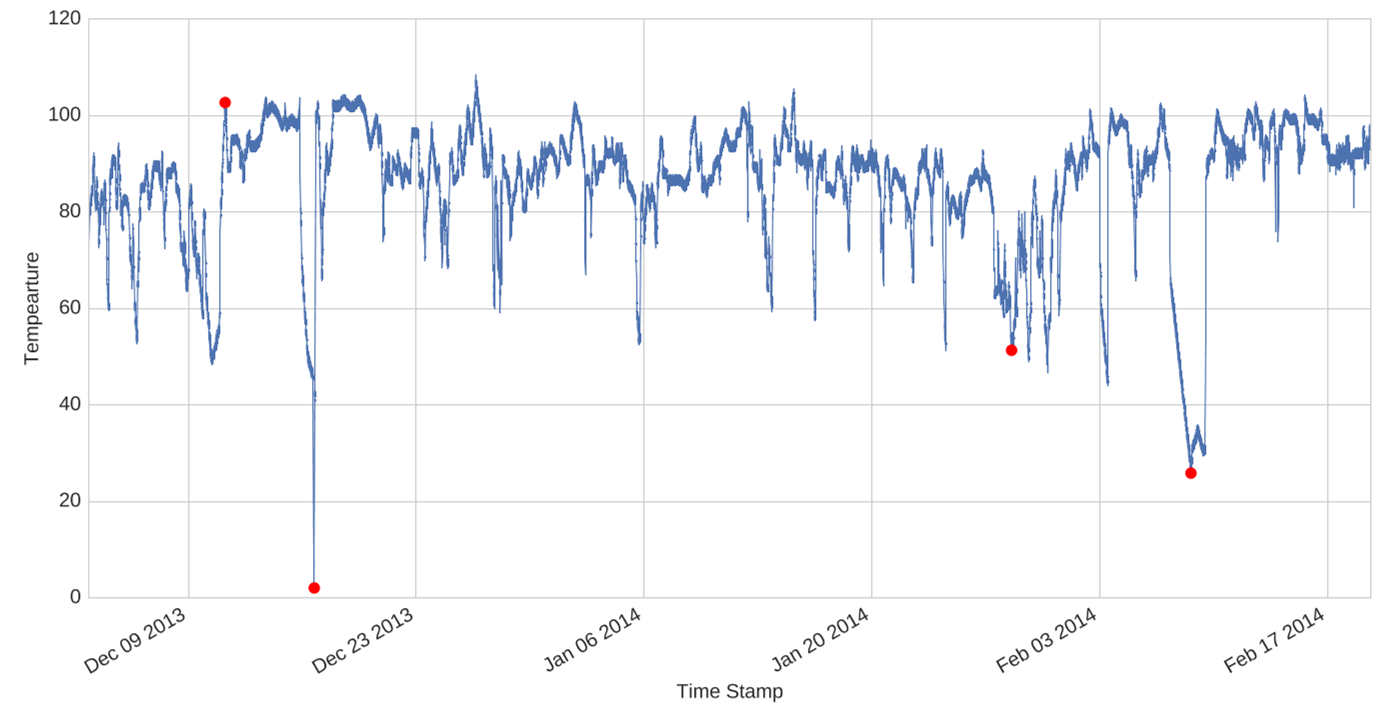


图3.1:Numenta的机器温度数据集。这个数据包含每5分钟的温度读数。有四种已知的异常由红色标记表示。x轴表示时间步长，y轴表示温度。

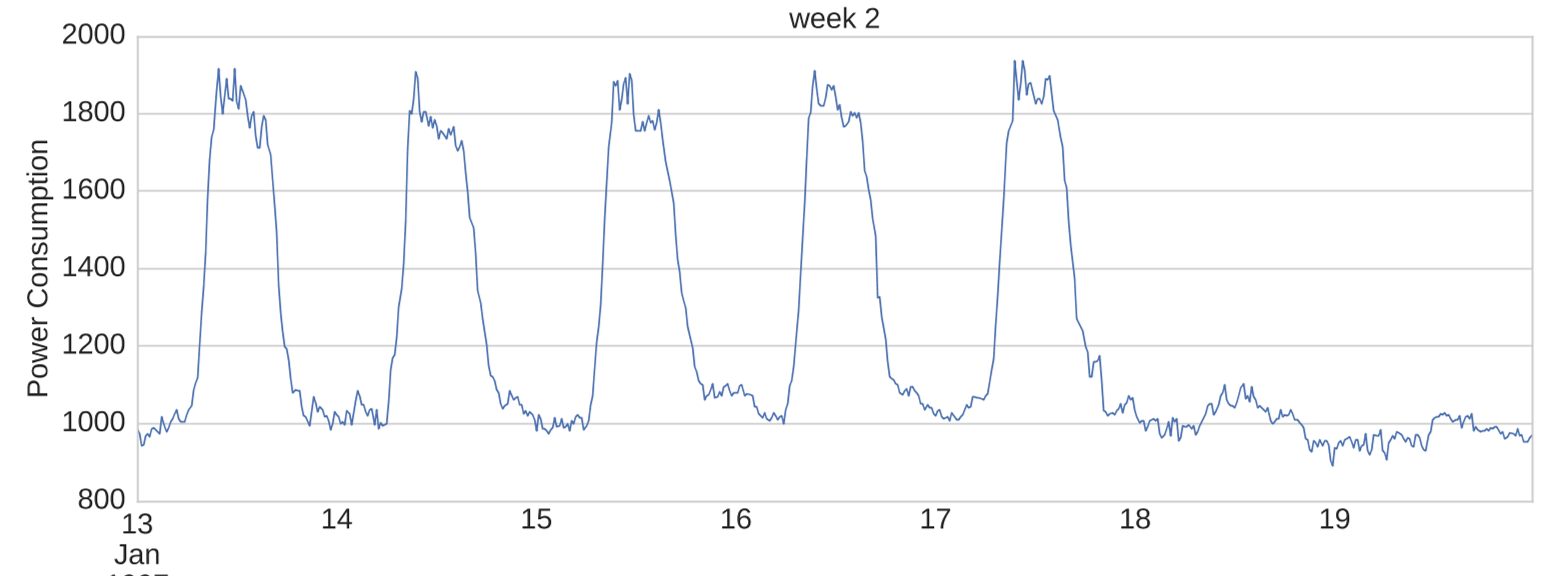
该数据集取自[39]，在Numenta的GitHub储存库中可用。数据集包含一个大型工业机器内部部件的温度传感器读数。这些数据是2013年12月2日至2014年2月19日期间的数据。每5分钟有22695次阅读。已知的原因有四种。数据如图3.1所示，异常以红色表示。第一个异常是计划关闭，第四个是灾难性的失败。另外两个异常在视觉上并没有明显的区别。

3 https:/ /github.com/numenta/NAB/tree/master/data

24

第三章。方法和数据集

3.3.2电力需求数据集



(a)周末有高需求的周循环和工作日的低需求。

(b)周一和周四出现异常的一周需求较低。

图3.2:电力需求数据集。(a)显示典型的一周(周一至周日)。(b)有异常的一周。x轴显示日期，y轴测量功耗。每15分钟录制一次。

第二个数据集记录了1997年荷兰研究机构的电力需求。这些读数每15分钟进行一次，总共进行了35040次观察。该数据每周循环周期为672次，有5个峰值和2个低，对应于工作日的高功率消耗和周末的低功耗。数据集以前在[31], [40]和[41]中使用过，在那里，低功耗的工作日被认为是异常现象。这些工作日与节假日同时发生。同样的，高功率需求的周末也可能被视为异常。我们使用相同的方法。图3.2中显示了正常和异常的日子。数据集可以从伴随[40]的webpage4下载。

方4 http:/ /www.cs.ucr.edu/ ~ / /不符

25

第三章。方法和数据集

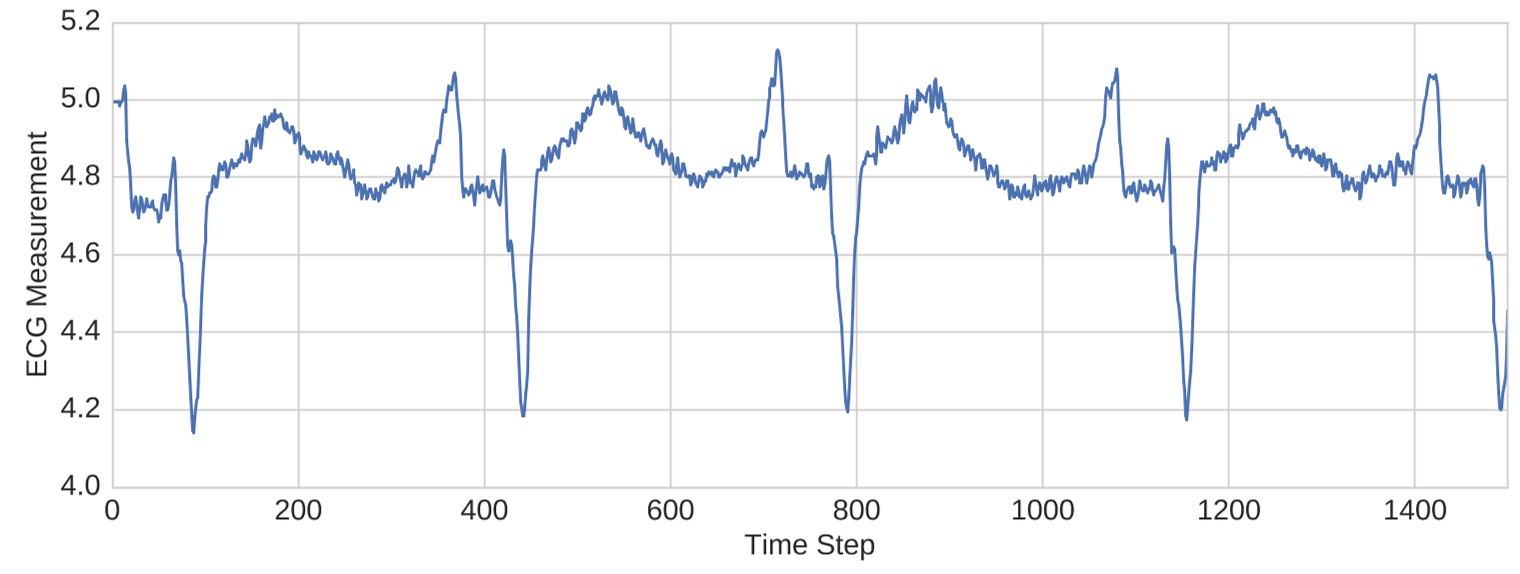
3.3.3心电图数据集

心电图是记录心脏电活动的时间序列。在物理银行的archive5中可用的心电图数据集被用于[31], [40], [41]和[42]等异常检测。在本报告中，我们使用了[40]中使用的数据集，因为它提供了由心脏病学家标注的标记异常。图3.3a显示了显示正常模式的数据集片段。共有18000份不同类型的异常数据。在[40]中被识别为最不寻常序列的三个被标记的异常，如图3.3b和3.3c所示。虽然这个数据集有一个重复的模式，但是模式的长度是不同的。

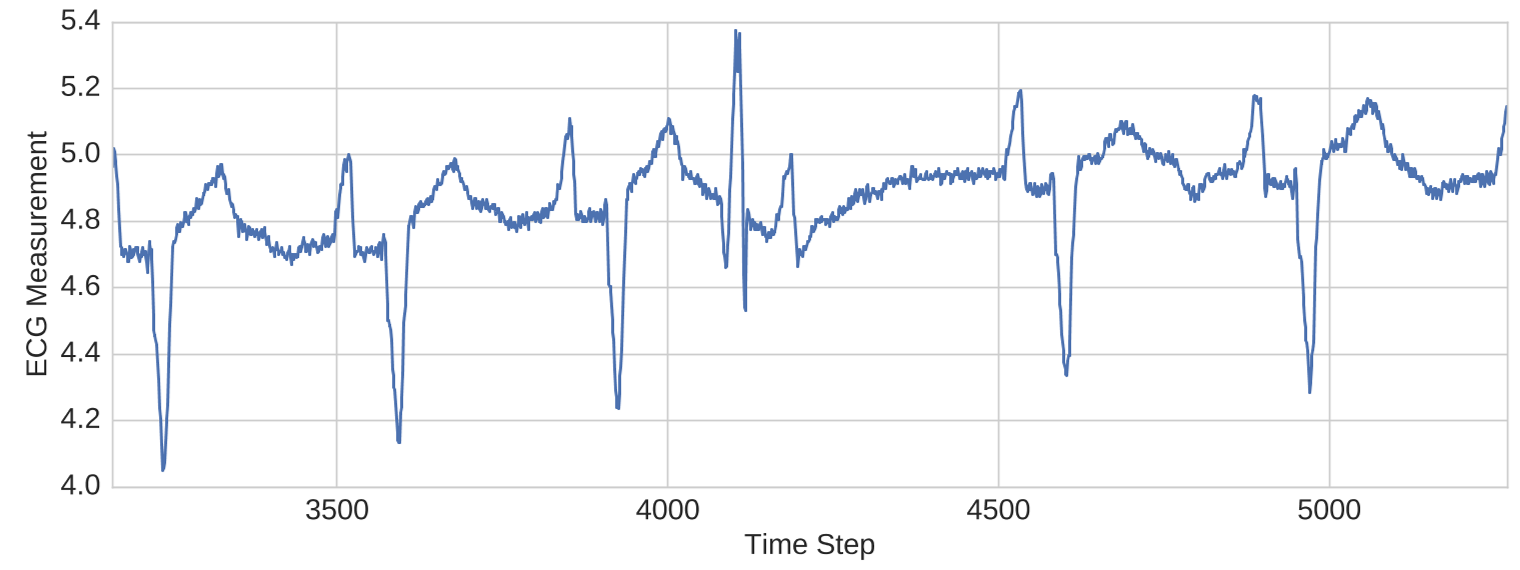
5 https:/ /physionet.org/physiobank/database/

26

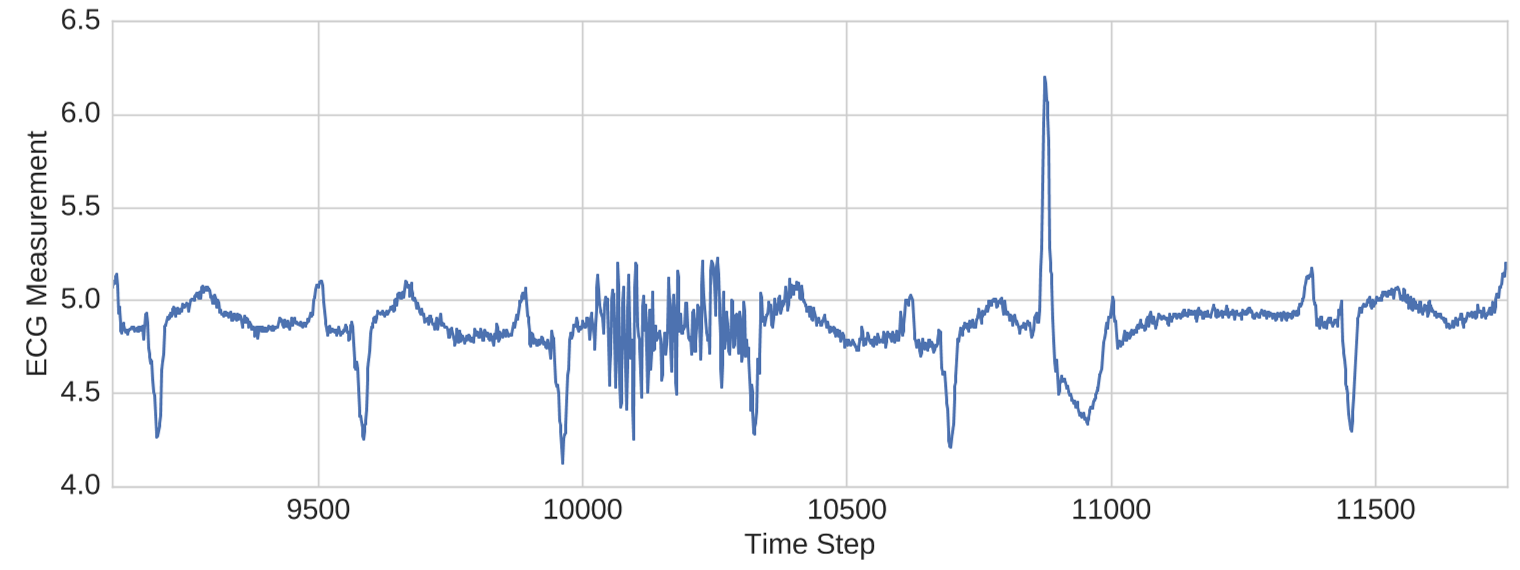
第三章。方法和数据集



(一)



(b)



(c)

图3.3:心电图数据集。(a)显示正常的心跳模式。(b)显示第一个异常。(c)显示另外两个异常。x轴表示时间步长，而y轴表示ECG测量值。这些数据显示了数据集的片段，使正常和异常的心跳很容易看到。

27

第四章

实验和结果

4.1主要结果

本节给出了每个数据集异常检测的主要结果。表4.1总结了用于不同数据集的预测模型。

以下4.4.1数据预处理

对于每个数据集，异常被划分为集合VA和T。然后用正常数据扩充这些集合。具有重复周期的数据集以这样的方式划分，使周期保持不变。剩余的数据被划分为集合N和VN。我们还将数据归一化为零均值和单位方差。训练集的均值和标准差用于其他集合的标准化。最后，将每个集合转换为算法所需的格式。所以每个输入样本都包含了回溯数的时间步骤。

4.1.2 Numenta机器温度数据集。

我们将4个异常值平均分成了VA和T，每一个都有大约20%的数据。其余数据用于训练预测模型。

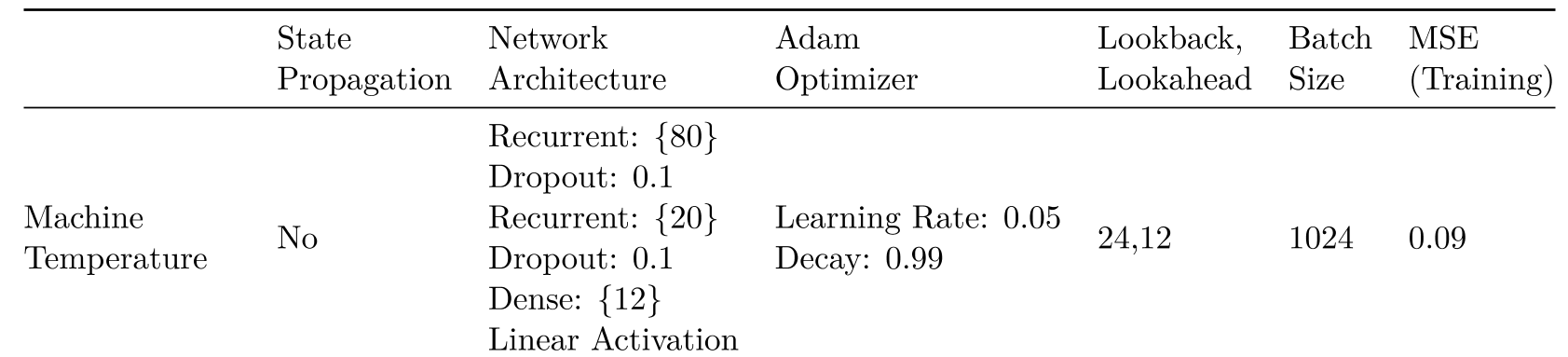
模型细节:LSTM RNN使用的是24的lookback, 12个隐藏的重复层，分别有80个和20个LSTM单元，一个有12个神经元的稠密输出层，以及一个0.1的退出。我们使用了Adam优化器的学习速率为。05，衰减为0.99，批量大小为1024。对200个早期停止训练的人进行了训练。由于数据不包含任何重复的模式，所以我们没有在批量之间维护LSTM状态。这个模型在N上给出了0。09的MSE。使用设置VA−11的阈值设置日志PD的值。然后使用set T来评估阈值。

评价:分别在图4.1和图4.2中分别给出了对集合VA和T的异常检测结果。−11的阈值是必要的检测

28

第四章。实验和结果

表4.1:每个数据集使用的预测模型的详细信息。网络体系结构指定了层的类型以及每个层中的单元数，以及退出值。在输出层中使用了线性激活。状态传播是指批量间的LSTM状态维护。



心电图没有

复发性:{ 60 }辍学:0.1

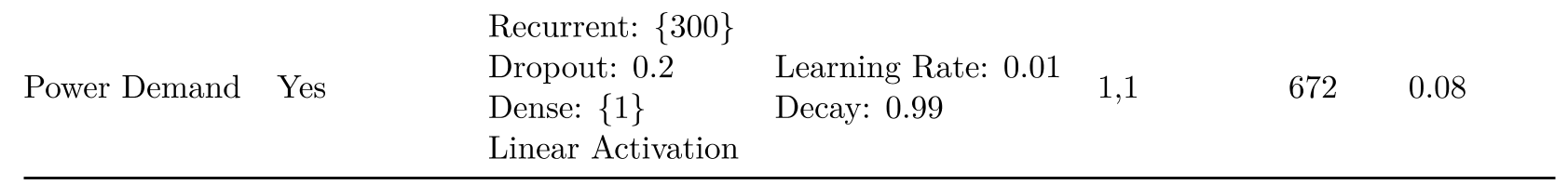
复发性:{ 30 }辍学:0.1

密度:{ 5 }

线性激活

学习速率:0.1

衰变:0.998,5256 0.10。



第一个异常在集合VA，但发生了一些假阳性。的阈值T−11发现第二个异常,但没有第一个异常检测。

我们将LSTM算法的结果与[39]的结果进行比较，该算法使用基于分层时间存储器(HTM)的异常检测算法[43]。作者定义了固定大小的异常窗口，每个窗口以一个异常为中心。在一个窗口内的最早的检测被认为是一个真正的积极的，而任何在窗口外面的检测是假阳性。采用加权计分机制，对窗口内的早期检测具有较高的权重。HTM算法的结果如图4.3所示。HTM检测四种异常中的三种，并有五种假阳性。在第一个异常窗口中，图4.3c HTM在真正的异常发生之前进行了大量的检测。这个异常窗口覆盖了LSTM算法在图4.2中所做的假阳性检测。事实上，LSTM算法的大部分误报都映射到HTM使用的异常窗口。因此，基于窗口的检测方法可以提高LSTM算法的性能，并能以较低的假阳性率检测所有异常。但是，我们没有开发这种检测方法，因为具体的检测方法应该由用例和目的来决定。应该注意的是，HTM算法有一个明显的优势，即使用一个小的训练窗口(图4.3a中的紫色阴影部分)，而我们使用了70%的数据用于训练目的。

29日

第四章。实验和结果

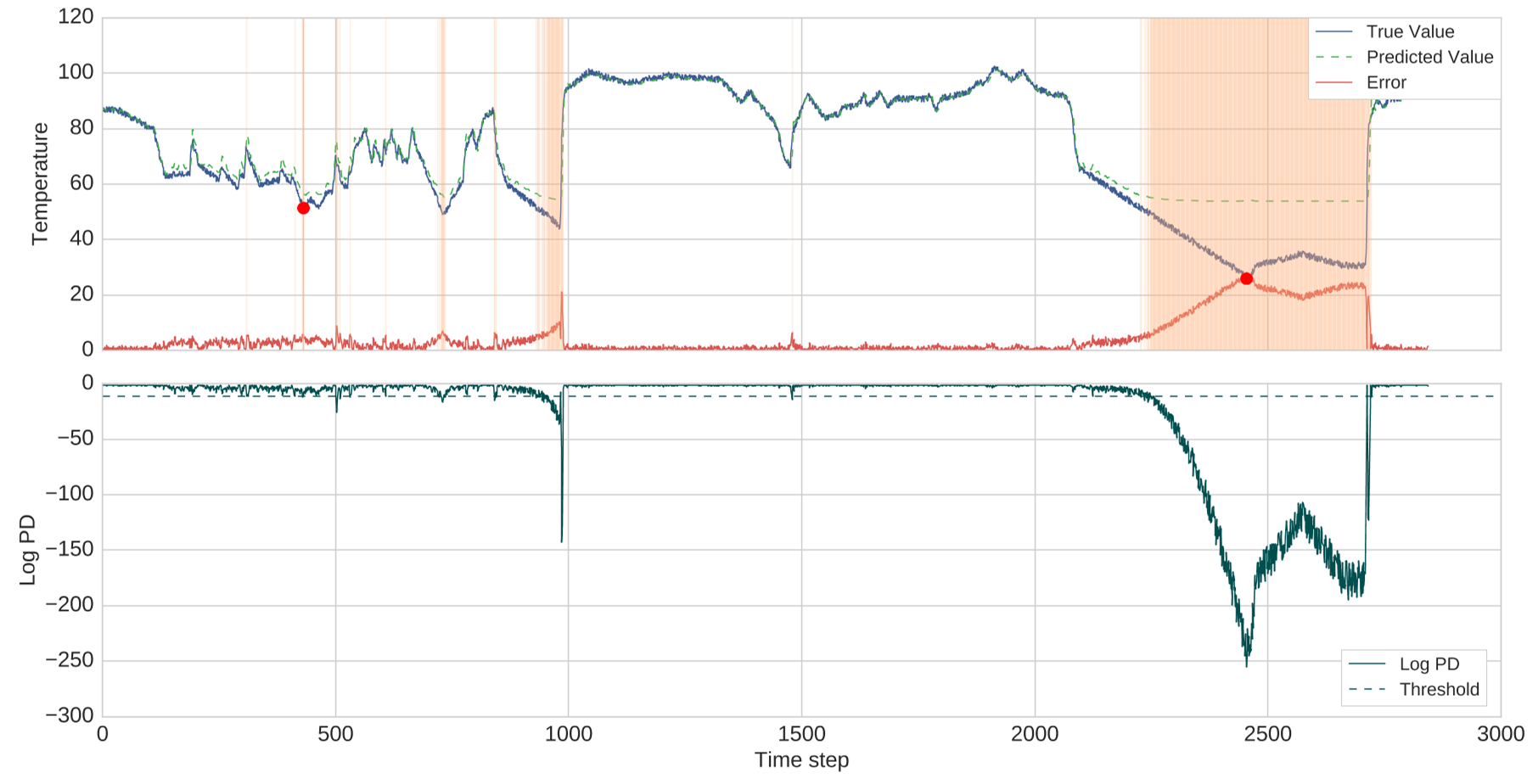


图4.1:机器温度数据集的验证集结果。上面的图显示了对VA的预测和相应的预测误差。图显示了日志底部PD的预测错误和阈值设定在−11。x轴表示时间步长，y轴有相应的度量值。红色标记突出显示了两个真正的异常。上图中阴影部分表示了LSTM算法的检测结果。

4.1.3电力需求数据集

数据集包含对应52周的读数。我们将这些数据分成了32周，VN由3周组成。剩下的几周分别在VA和T之间分配。有9个异常，8个异常对应于工作日的低需求，只有一个周末的异常需求高。在这9个异常中，有4个被分配给了VA，剩下的给了T。

模型细节:RNN由一个单一的重复层和300个LSTM单元组成，后面是一个密集的输出层，其中有一个神经元，以及一个0.2的退出。向后看，看前面等于1。我们对RNN进行了50个周期的训练，早期停止并使用了Adam优化器，学习速率为0.01，衰减为0.99，批处理大小为672。由于数据具有长周期，所以我们在批量之间维护LSTM状态。集合VA和T的结果分别如图4.4和4.5所示。预测模型在N上给出了0。08的MSE。利用VA的预测误差，设定了24小时的阈值。

30.

第四章。实验和结果

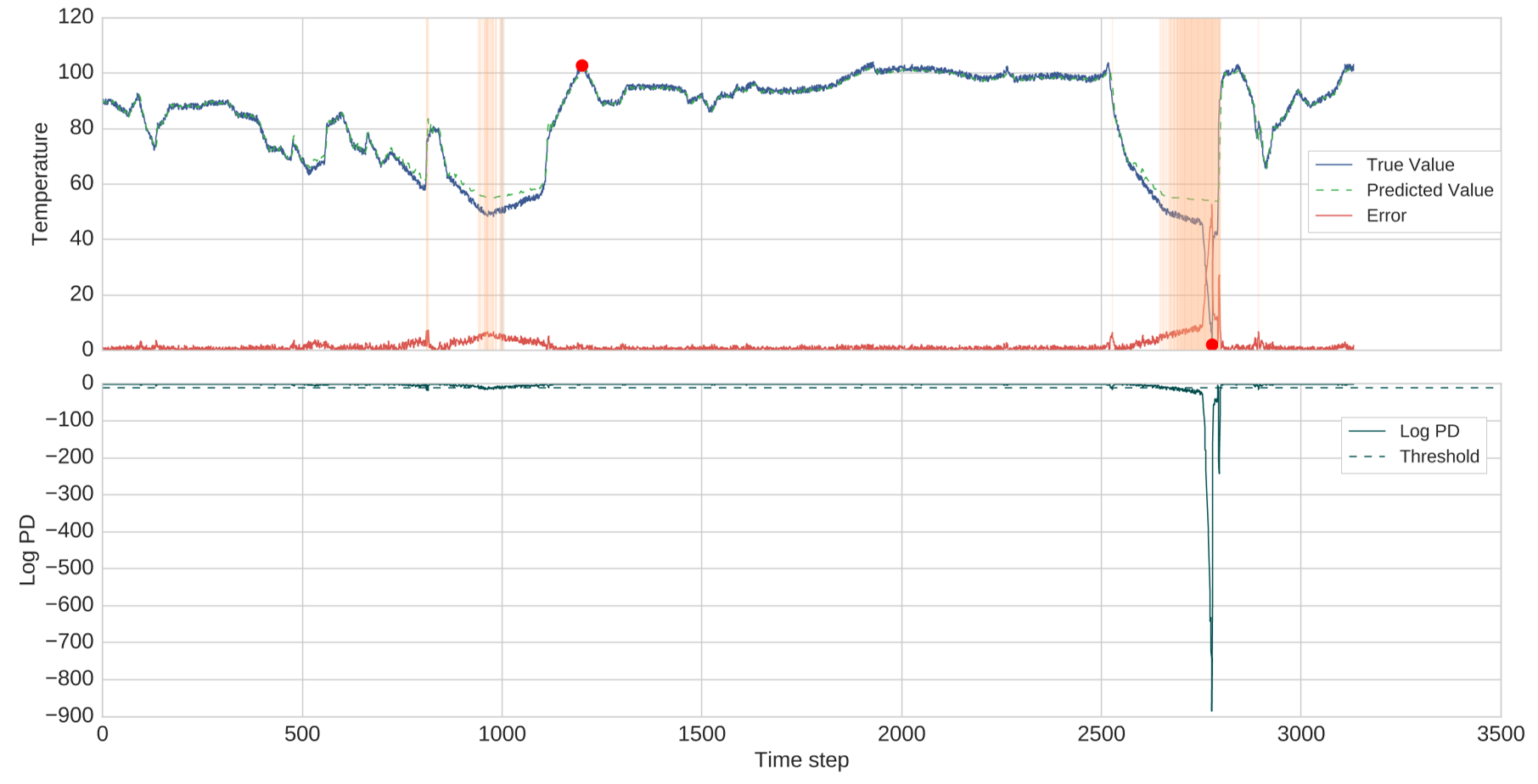


图4.2:机器温度数据集的测试集结果。上面的图显示了对T的预测和相应的预测误差。图显示了日志底部PD的预测错误和阈值设定在−11。x轴表示时间步长，y轴有相应的度量值。有两个真正的异常，显示为红色标记。顶部图的阴影部分是包含由LSTM算法产生的检测的区域。

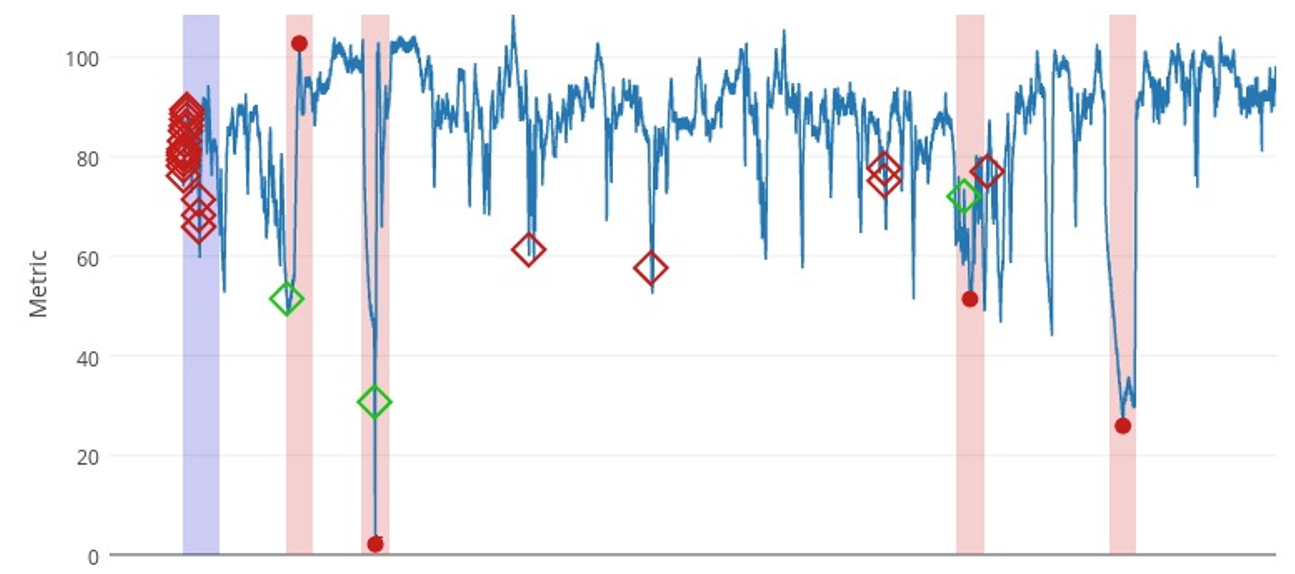
评估:电力需求数据集的结果如图4.4和4.5所示。24的阈值检测了集合VA中的所有4个异常，而不产生任何误报。在集合T中所有的5个异常都被检测到，但是有一个假阳性。仔细观察，这个假阳性结果是一个突然的峰值。该数据集已在[40], [41], [31]和[36]中使用。[40]找到了三个最不寻常的星期，所有这些都发生在两个星期的假期里。[31]和[36]使用LSTM网络，将整个星期的假期视为异常。他们都将F-score作为衡量绩效的指标。我们没有使用特定的度量标准来检测，因为这应该由用例和低数量的异常来控制。

4.1.4心电图数据集

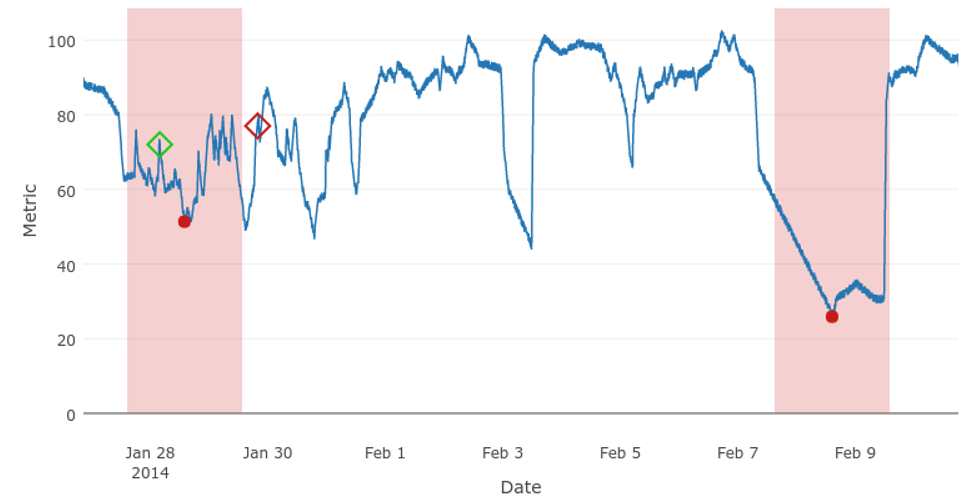
在ECG数据集中，18000个样本中只有三个标记异常。因此，我们不使用集合VA来寻找阈值。相反，数据集被划分为一个正常的训练集(N)，一个用于早期停止的常规验证集(VN)，以及一个包含所有三个异常的测试集(T)。

31日

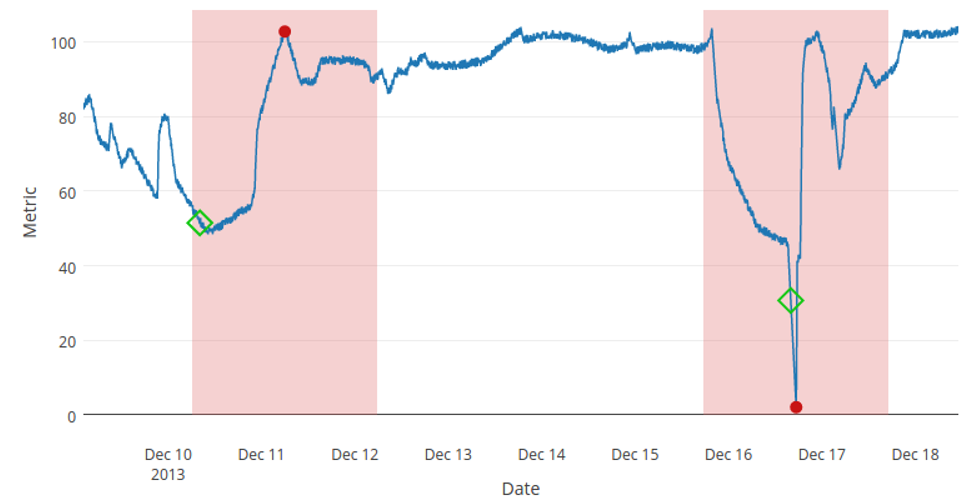
第四章。实验和结果



(一)



(b)



(c)

图4.3:HTM在机器温度数据上的结果。(a)显示整个数据集的结果。(b)显示与图4.1对应的结果。(c)显示与图4.2相对应的结果。x轴表示时间，y轴表示温度。真正的异常是用红色标记表示的。绿色/红色方块代表真/假阳性。粉色阴影部分是异常窗口。(a)中的紫色阴影区域是测试窗口。由NAB GitHub存储库中的代码生成的图。

32

第四章。实验和结果

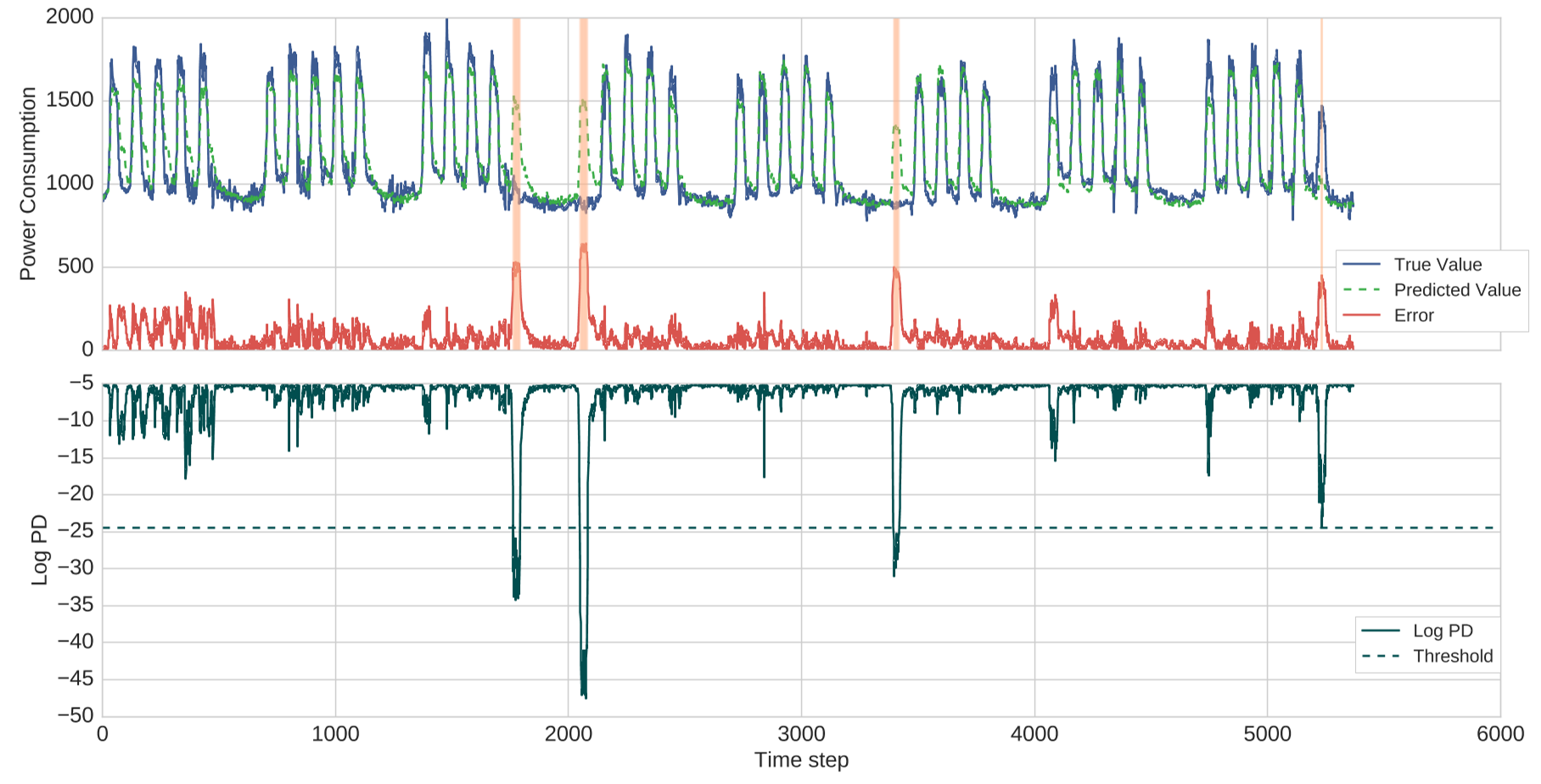


图4.4:电力需求数据集的验证集结果。上面的图显示了对VA的预测和相应的预测误差。底部的图显示了预测错误和阈值的日志PD值。x轴表示时间步长，y轴有相应的度量值。这些图包含8周的周期，每个周期分别为3、4、6和7。前三种反常现象对应的是需求较低的工作日，而最后一个异常是周末(周六)的需求旺盛。阈值设置在-24以检测所有异常。上图中阴影部分突出显示了算法检测到的异常区域。

模型细节:使用的预测模型是一个叠加的RNN，由两个隐藏的重复层组成，分别有60个和30个LSTM单元，一个稠密的输出层，有5个神经元，8个向后看，5个向后看，1个0。LSTM状态在批次之间不存在。我们使用Adam优化器训练了RNN，学习速率为0.1，衰减0.99，批大小为256。该模型接受了50个周期的早期停止训练。MSE的N是0.10和−23的阈值设置日志PD T值的预测错误设置。

评价:图4.6所示为set T的异常检测结果。−23的阈值模型检测所有三个异常。这些结果与[40]的结果相似，[40]在这一数据中找到了三个最不寻常的序列。此外，不一致的排序与日志PD值的顺序相匹配，其值较低，表明一个更不寻常的序列。

33

第四章。实验和结果

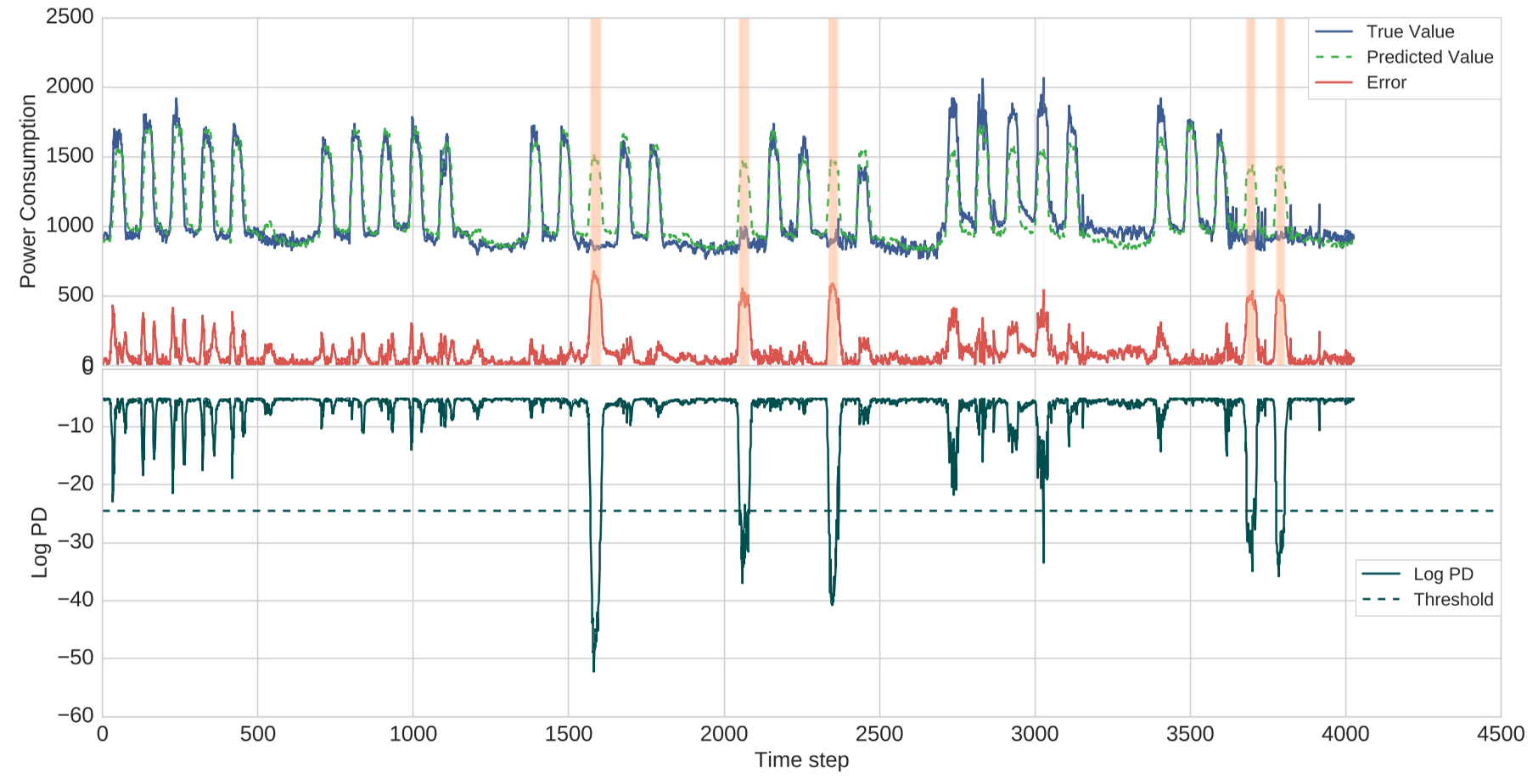


图4.5:电力需求数据集的测试集结果。上面的图显示了对T的预测和相应的预测误差。图显示了日志底部PD的预测错误和阈值设定在−24。x轴表示时间步长，y轴有相应的度量值。这些地块包含6周周期，周期为3，周期为1，周期为4和6。由算法检测到的异常由阴影区域表示。所有5个异常都是通过设定阈值检测出来的，但有1个假阳性。

4.2维护LSTM状态

使LSTMs从数据中学习有用模式的一个重要因素是对LSTM状态的处理。我们用不同的方法来构建LSTM状态及其对预测和检测性能的影响。我们将在本节讨论结果。

机器温度数据集似乎没有任何重复的模式，而且模型的性能也被发现对状态的维护不敏感。ECG数据集有一个重复的模式，大约有370个时间步骤，尽管实际的长度略有不同。我们期望在批次之间保持状态会得到更好的结果，但是结果是相似的，不管我们如何处理LSTM状态。我们相信，即使ECG数据集有一个很长的模式，相关的时间关系也只是在最近的几个时间步骤中出现。

34

第四章。实验和结果

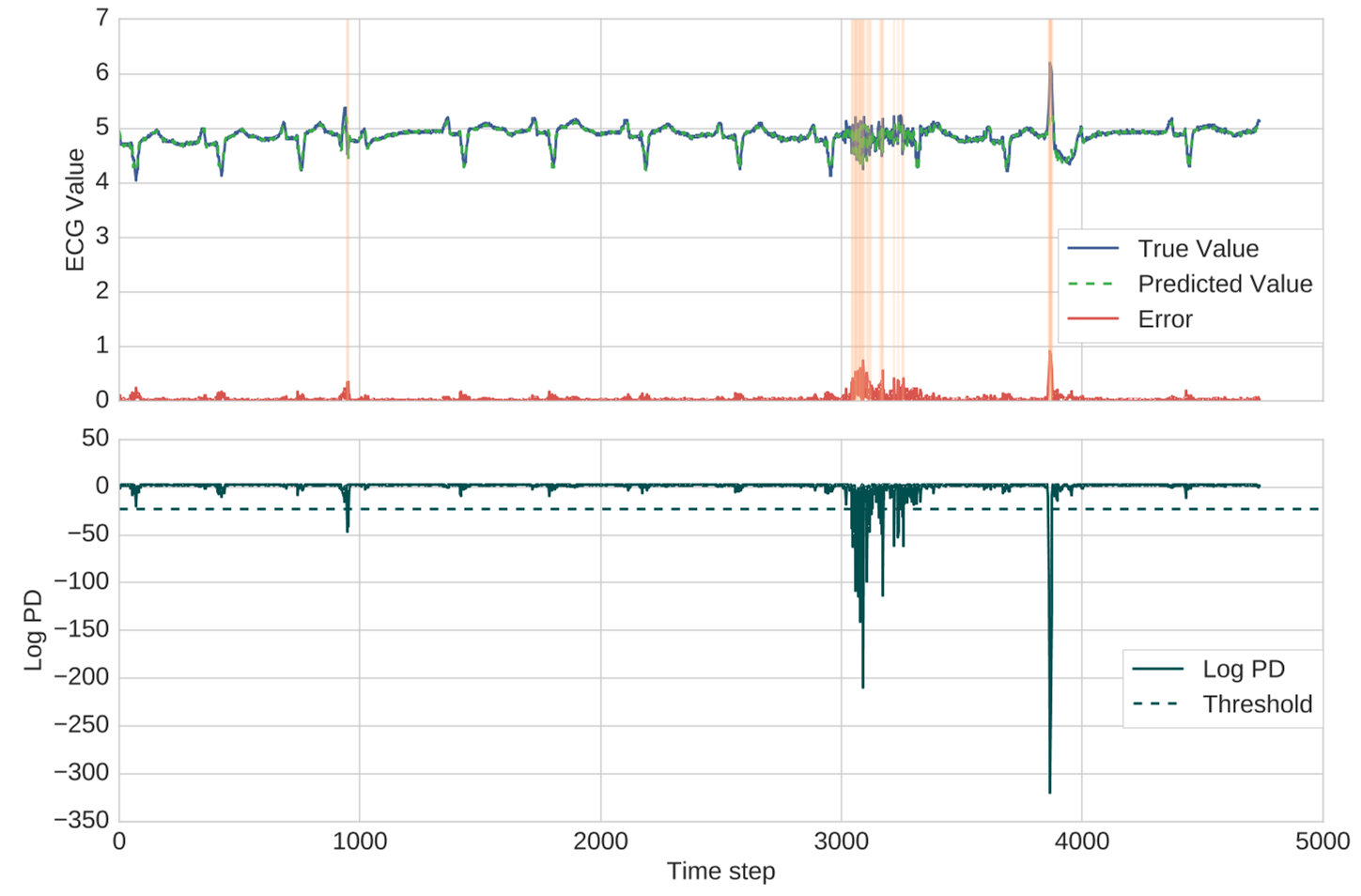
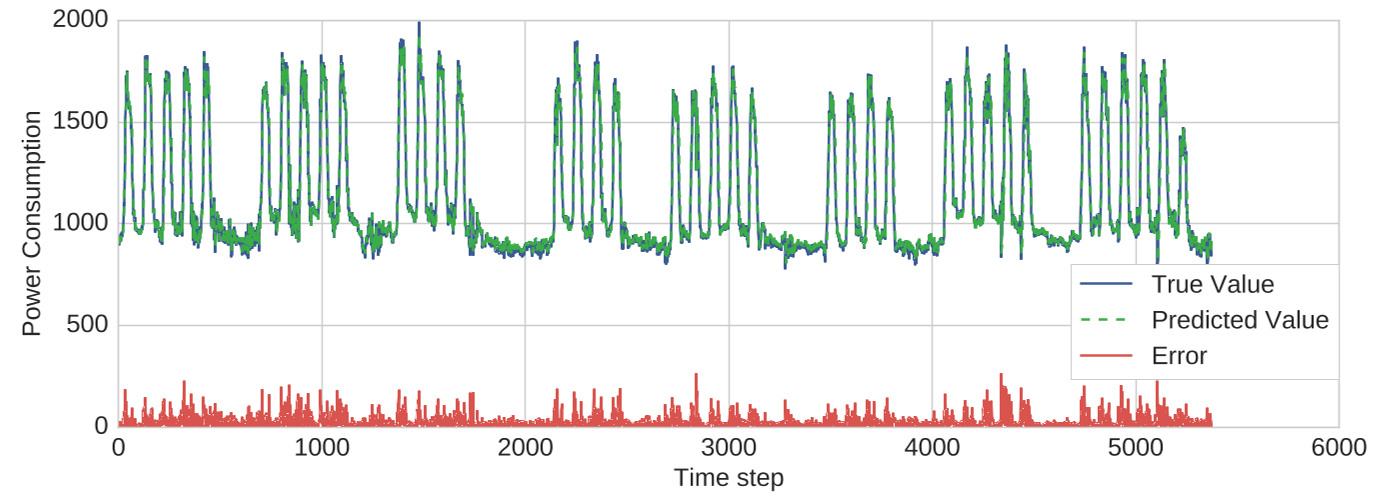


图4.6:ECG数据集的结果。上面的图显示了对T的预测和相应的预测误差。底部的图显示了预测错误的日志PD值和在23上设置的阈值。x轴表示时间步长，y轴有相应的度量值。有三个异常。由算法检测到的异常由阴影区域表示。

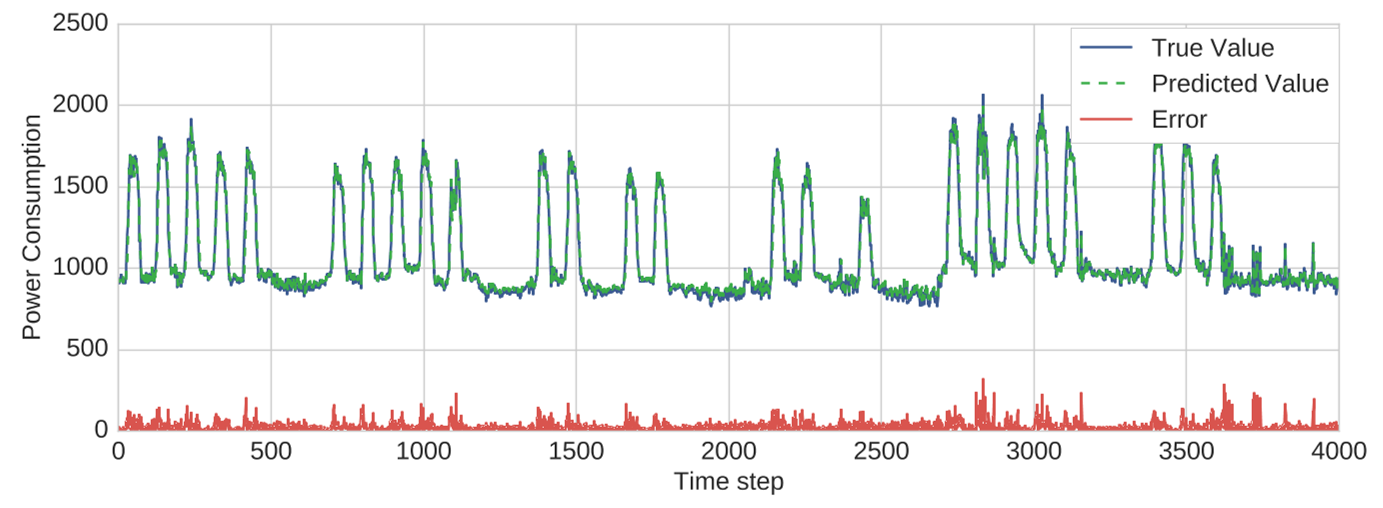
对于电力需求数据集的状态维护是获得理想结果的关键。我们首先尝试了默认模式，即在每批处理之后重新设置状态。在所有其他方面，预测模型与第4.1.3节相同。该模型的MSE为0.0283。更重要的是，在集合VA上的MSE和有异常的T分别为0.017和。019(这与第4.1.3节的结果相比，分别是在N、VA和T上的MSE分别为0.08、0.15和0.26)。虽然这些结果表明预测有显著的改善，但它们对异常检测没有帮助。对于异常检测工作，我们要求在集合N上有一个低的MSE，在集合VA和T上有一个更高的MSE。这是因为集合VA和T包含异常，模型应该对异常数据有较高的预测误差。使用默认模式的结果(不包括批量的状态维护)如图4.7所示。很明显，这个网络没有学会每周重复5个峰值和2个低点的模式。例如，在图4.7a中，第三周周期只有4个高峰和周五的低点，但是模型预测遵循实际的(反常的)观测，而不是预测周五的高功率消耗。因此，结果预测误差不提供异常和正常数据点之间的差别。

35

第四章。实验和结果



(一)



(b)

图4.7:无维护状态下的电力需求数据预测。(a)显示VA和(b)的结果，显示T的结果。由于没有维护状态信息，模型无法学习周模式。x轴有时间步长，y轴表示功耗。

在更仔细的检查中，我们发现电力需求数据在不同的时间尺度上有多个模式:有一个包含672个时间步骤的很长的每周模式，但是也有96个时间步骤的日常模式。为了了解这些不同的模式，LSTM状态需要建立对过去信息的记忆，并提供一个上下文，使模型能够根据一周的天数做出正确的预测。但是，在默认模式下，每个批处理后，状态都被清除。在没有任何有用的状态输入的情况下，模型学习几乎完全受到最近观测的影响，而权重的调整是为了最小化预测误差。这使得模型能够适应异常的数据，因为它尽可能地跟随最近的观测。

接下来，我们尝试了Keras提供的有状态模式。该方法用于第4.1.3节中给出的最终模型。在所有的训练批次中，国家被维持，并且只有在训练数据的新时期开始时才复位。这使得模型可以构建整个过程的状态。

36

第四章。实验和结果

培训数据贯穿所有培训批次。虽然在每个时代之前，状态都被重置，但是模型的权重仍然在不同的时代中进行调整，LSTM门学会在细胞状态中存储相关的激活。因此，国家为模型提供了必要的输入，以学习不同的模式并做出正确的预测。

在有状态模式中还有一些其他有趣的观察。首先，我们发现性能对批处理大小非常敏感。批量大小约等于周周期的长度是学习远程模式最有效的方法。理想情况下，LSTM模型应该能够学习模式，而不考虑批量大小。第二，即使在每个时代之前，国家没有重置，网络的性能也是相似的。通常建议在每个时代之后重新设置LSTM状态，因为状态可能会继续增长并变得不稳定。然而，由于我们使用的数据集非常小(20000到40000个版本)，而且由于特定的Keras的构造，很难下结论。

4.3使用固定时间窗的前馈NNs。

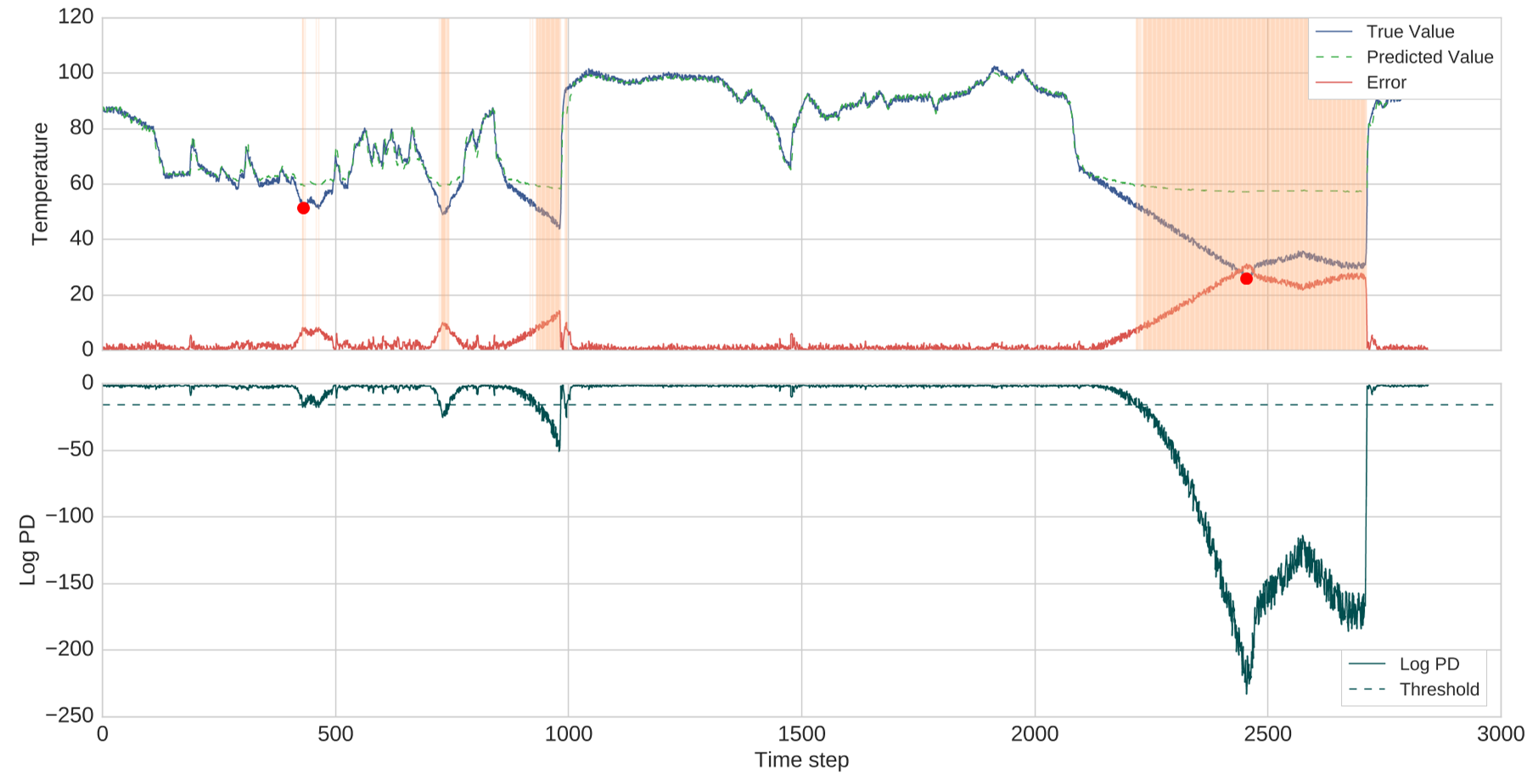
我们注意到，仅对电力需求数据集需要跨批次的状态保存。对于其他两个数据集，仅对单个训练样本的回溯数进行维护。我们仍然得到了合理的预测和异常检测结果。这使我们怀疑，如果使用像LSTM RNN这样复杂的模型获得任何好处，如果使用固定大小的时间窗口的前馈NNs也能捕获时间关系。我们用前馈神经网络作为预测模型进行了实验。

图4.8a和4.8b分别显示了机器温度数据集的集合VA和T的结果。这里使用的预测模型有两个隐藏的层，每个层有32个神经元，有sigmoid激活，没有退出。该模型使用了Adam优化器，使用了0.01的学习速率，批量大小为1024，并提前停止。所有其他细节，包括输出层、lookback和lookahead都与LSTM模型相同。该模型接受了100个新纪元的训练，并在set N上进行了MSE。−16的阈值设置使用VA和给类似的结果比LSTM异常检测模型。

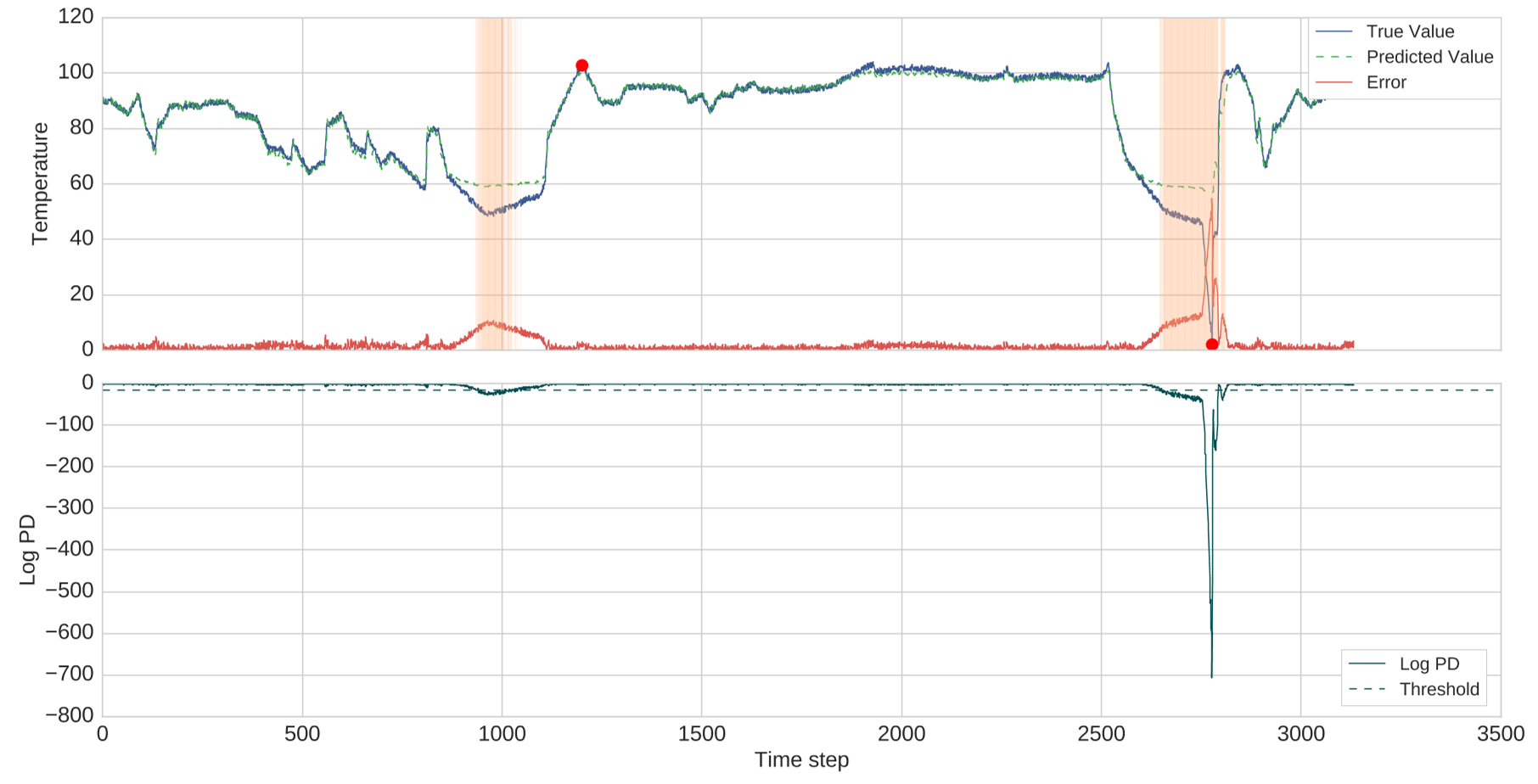
对于ECG数据集，NN使用了两个隐藏的层，分别有64个和32个神经元，有sigmoid激活，没有退出。该模型接受了Adam优化器的培训，使用的学习速率为.001，批量大小为1024，并且提前停止。所有其他参数与前面相同。该模型接受了80个周期的训练，并给出了0。07的MSE。在日志−25 PD阈值T检测所有错误的异常。ECG数据集的结果如图4.9所示，与LSTM模型相似。

37

第四章。实验和结果



(一)



(b)

图4.8:在机器温度数据上使用前馈神经网络的结果。(a)显示了集合VA的结果。(b)显示了集合T的结果。−16的阈值设置使用VA和结果类似图4.1和4.2。x轴表示时间步长，y轴表示相应的度量值。

38

第四章。实验和结果



图4.9:在ECG数据上使用前馈NN的结果。T和一个阈值集的结果−25检测3异常。x轴表示时间步长，y轴表示相应的度量值。

对于电力需求数据集，前馈神经网络给出了良好的预测精度，但没有发现任何异常。我们尝试了不同的架构和窗口大小达到100。这与LSTM算法遇到的问题类似，因为我们没有在批处理之间保持状态。NN未能了解数据中存在的长且多模式。

4.4其他实验

4.1.1 Lookback效果

参数lookback是RNN为反向传播而展开的时间步数。对于RNN的展开，没有推荐的值，而最优值取决于数据的性质和它存在的时间相关性。从8到200的Lookback值被成功用于不同的任务。虽然lookback值是影响学习的一个重要参数，但是LSTM RNNs仍然可以学习比lookback长度更长的模式，因为LSTM状态可以存储来自任何前一个时间步骤的信息。在[44]中显示，在单时间步上进行反向传播训练的RNNs能够为语言建模任务学习更长的模式。我们发现

39

第四章。实验和结果

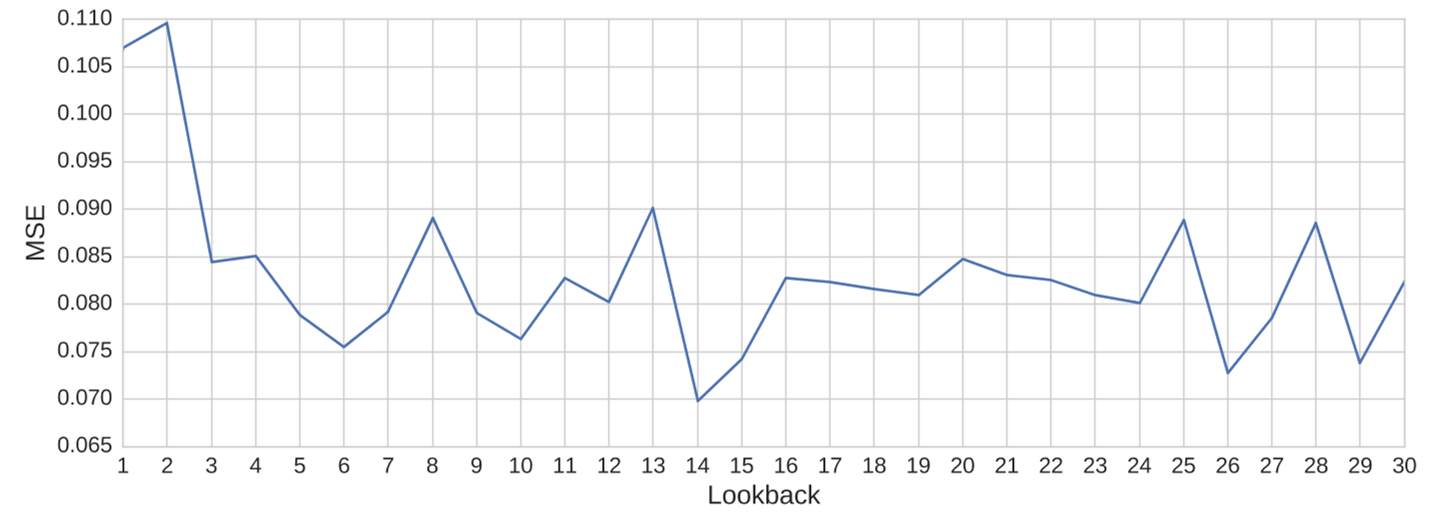


图4.10:回顾和预测精度。图中为不同的lookback值对ECG数据集进行MSE的训练。在14之后增加lookback值不会减少MSE。x轴显示回视值，y轴具有相应的MSE值。

增加回溯值可以使预测性能提高到一个极限，并且通过进一步增加回溯值来获得改进。图4.10在单个时间步预测任务(lookahead = 1)中，对ECG数据集进行了MSE的训练。MSE初始值最小为0.07，而向后的值为14。在14之后增加lookback值不提供任何进一步的改进。我们在其他数据集上也发现了相同的行为。

4.4.2预测精度与异常检测。

在我们的一些实验中，我们经历了预测性能与异常检测结果之间的权衡。为了使预测最小化，RNN对异常检测并不是最好的。在整个集合VA(和T)中，预测误差很小，因此很难找到一个阈值来分离正常点的异常，同时又不会产生太多的误报。这种行为的一个实例是当我们没有在批次之间保持状态时，在power demand dataset上。这在第4.2节中更详细地描述。

对这种现象的直观解释如下。当我们对预测模型进行优化时，由于MSE在N和VN上的持续下降，导致无法提前停止的一种过度拟合。模型的权重是调整的，模型预测只受最近的观测和模型的影响，不考虑任何时间关系或来自其状态的信息。因此，该模型试图拟合异常观测值，使预测误差较小。模型训练对异常和模型没有监督，没有目标或反馈来防止异常的拟合。我们发现的唯一补救办法是手动调整模型，以检测set VA的异常情况。然而，它是以增加MSE值为代价的。手动调优主要涉及降低一个或多个的值。

40

第四章。实验和结果

下面的参数:向后看值，在重复层中LSTM单元的数量，或重复层数。

41

第五章

讨论

5.1异常检测数据集、指标和评价。

我们在项目中遇到的最大的挑战是标签数据集的选择。异常检测被认为是一个重要的问题，并被广泛研究。因此，在异常检测研究中，标记和基准数据集的可用性仍然是一个主要问题，这是令人惊讶的。处理这个问题的一种方法是使用已知的只有正常观测值的数据集，通过在数据点添加随机噪声来引入人工异常。但是这种方法存在一些问题。首先，在人工异常的情况下，很难对该算法对真实世界数据的有效性有任何想法。第二，如果有足够的标记异常，最好使用监督分类方法，这比无监督异常检测要容易得多。此外，在现实世界中，异常现象是罕见的事件，可以认为发展过程也应该反映这一事实。

我们决定不使用诸如F-score、precision、recall等指标来评估我们的方法。我们认为，考虑到异常的稀缺性，使用一个指标来表现出良好的表现将是一种误导。由于仍然有必要证明我们的方法的有效性，我们选择将我们的结果与以前的作品进行比较。虽然我们知道我们使用的每个数据集都有很少的异常，但我们的目的是为了显示LSTM对时间异常检测的适用性，而不是提出一种更好的算法。

一个相关的决定不是开发一种特殊的异常检测方法，而是使用一种阈值方法来区分异常和正常的观测。开发一种特殊的异常检测方法，应受到预期目的和异常检测领域的影响。例如，在电力需求数据集的情况下，可以有多种方法来定义一个异常:一天可能是一个异常，整个星期可以被视为异常，或者是一个异常。

42

第五章。讨论

甚至可以将个别观测定义为异常，进行实时检测。ECG数据集具有点异常和集体异常。机器温度数据集带有带注释的点异常。但很明显，围绕这些点的所有地区都是不寻常的。HTM算法利用这些点周围的窗口进行异常检测。我们使用的阈值方法可以用于点异常检测，无需太多修改。但是对于许多应用程序来说，失败之前有不同寻常的序列。在这种情况下，集体异常检测方法更有用。类似于[33]的一种可能的方法是，定义一个滑动窗口来预测错误，并在窗口中使用错误和来计算异常值。

在本报告中，我们对异常检测研究中的一些主要问题进行了评论，包括对算法的评估。由于不同的算法在异常的性质上存在差异，并且有不同的取舍，因此评估的问题更加复杂。这使得将不同的算法进行比较是徒劳的。已经有人试图解决这些问题。在[38]中提出了从分类数据集创建异常检测数据集的标准和过程。在[2]中，作者研究了不同的度量标准和数据集，这些数据集被用来进行离群评估。他们强调需要通过研究社区采用共同的基准。根据作者的观点，任何新的离群检测方法都应该得到支持，通过在各种各样的数据集上演示其性能，包括几个指标，以及各种各样的参数。然而，在实践中，通过在非常特定的数据集上对其进行评估，并只使用几个度量标准，就可以获得一种新的高级算法。作者在20多个具有不同特征的数据集上展示了12种标准算法的结果，并规定使用这些算法来评估新的算法和数据集。他们还提到了现有研究中缺乏可再现性。虽然这两种方法都不能提供异常检测的基准数据集，但我们认为它们代表了为异常检测评估开发许多需要的共同标准的重要步骤。

5.2正常的假设

我们做的一个假设是残差或预测误差的正态分布。我们对每个数据集的残差进行了Shapiro-Wilk测试。根据测试结果，我们拒绝了预测错误通常分布的假设。对于所有的数据集，这些错误都有肥尾，并且有轻微的偏移。然而，由于我们只使用正态假设来获得判别异常的阈值，该算法仍然可以在没有任何变化的情况下使用。

43

第五章。讨论

5.3 LSTMs vs前馈NNs。

从我们对LSTMs的分析和我们进行的实验来看，有几点值得讨论。最紧迫的可能是前馈NNs与LSTM RNNs的性能。在这三个数据集的两个数据集上，用固定大小的时间窗口的前馈NNs对LSTMs的时间序列预测和异常检测给出了相似的结果。与前馈神经网络相比，LSTMs工作的理论和数学更复杂，更具有挑战性。培训和调优LSTM网络也是一个非常耗时且容易出错的过程。这就提出了一个问题，如果使用LSTMs对前馈NNs有什么好处。

前馈神经网络对过去没有任何明确的记忆。它们对时间关系建模的能力受到输入时间窗口大小的限制。即使对于特定的时间窗口，输入数据也被视为多维特征向量，而不是作为观察序列。该方法只适用于没有远程时间依赖的简单时间序列，而且任务的所有相关信息仅包含在最近的几个输入中。利用后见之明，可以推断出机器温度和心电图数据集构成了这样一个简单的时间序列。视觉上这两个数据集是不同的。虽然ECG数据集有重复的370次时间步骤，但是机器温度数据集似乎没有任何模式，而且看起来是相当随机的。

在复杂时间序列模型中，电力需求数据集的结果具有更大的意义，因为它们证明了LSTMs在模拟复杂时间序列中的优越性。电力需求数据集不仅有一个长重复模式(670个时间步)，而且在不同的时间尺度上也有不同的模式:一个日模式，一个周模式。在使用前馈神经网络时，我们不能得到任何令人满意的异常检测结果。仔细地维护LSTM细胞状态在学习相关模式和检测异常方面是至关重要的。

在[45]中发现了一些对我们的推理的支持，作者将LSTMs应用于两个基准时间序列数据集的预测。在他们的实验中，LSTM模型只能访问当前的时间步，而前馈的NNs则得到连续时间步的窗口。研究发现，对于特定的数据集，LSTM模型的表现优于前馈神经网络。作者的结论是，对于许多时间序列的任务，只需要最近的几个时间步骤，而LSTM能够长时间地记住过去的信息并不能带来任何好处。他们建议只在基于时间窗口的方法不提供所需的结果时使用LSTMs。并提出将两种方法结合起来，首先训练前馈神经网络，然后利用LSTMs进一步降低预测误差。

44

第五章。讨论

要知道数据中存在的那种关系是很难的，或者如果数据需要像LSTM那样复杂的模型。通常，在转移到复杂模型之前，最好先测试更简单的模型。奥卡姆剃刀原理和其他科学一样适用于机器学习。在这个项目中，其中一个目标是研究和探索LSTMs。只有在我们的LSTMs实验之后，我们才选择使用前馈神经网络。然而，在使用LSTMs时，似乎没有尝试更简单的方法。这在很大程度上归因于最近LSTMs的流行以及它们在许多连续的和时间问题上的成功。例如，[33]和[36]使用LSTMs进行时间序列异常检测，但不将其与更简单的方法进行比较。

5.4时间异常检测的LSTMs。

我们用三个不同的数据集进行实验。这些数据集不仅存在于它们所具有的模式中，而且也与异常的性质不同。机器温度数据集带有标记异常。它没有任何重复的模式，而且看起来是随机的。有两个异常突出，是极端值。另外两个似乎很随意。在[39]中，有人提到，后一种异常是相当微妙的，并且之前有一些偏离正常模式的小偏差，这种模式在视觉上是无法探测到的。作者强调了异常窗在检测这些异常时的效用。在我们的实验中，LSTM算法对这两个异常进行了检测，虽然它没有检测到实际的异常。然而，正如讨论基于窗口的方法一样，将会得出类似的结果[39]。

ECG数据集也有注释异常，包含极端值和异常序列。它有一个重复的模式，尽管模式的长度有所不同。我们的方法检测到所有三个异常。功率需求数据集具有最长的依赖关系，在不同的时间尺度上显示模式，并包含噪声。虽然没有标签，但我们可以把这周作为背景，我们可以区分正常和不寻常的模式。通过仔细地维护LSTM状态，模型学习了多个模式，并检测到所有异常。此外，异常分数也能很好地说明异常的范围或概率。

基于我们的实验，我们认为LSTMs是有效的时间异常探测器。我们可以对不同特征的数据集进行LSTMs的调优，并检测各种类型的异常。LSTMs的内存允许它们在进行预测时使用上下文和输入数据。尽管我们经历了一个LSTM网络，它的预测精度并不一定是一个好的异常探测器。关键是利用LSTM内存使其保留相关的过去信息。

45

第五章。讨论

我们使用LSTMs进行了多次步进预测。对于异常检测，一次步进预测就足够了。为了展示LSTMs作为时间序列模型的能力，我们做了多次的时间步预测。预测多个时间步骤可以提供异常行为的早期指示，并作为从异常检测到异常预测和可能预测维护的重要第一步。

5.5 LSTMs:进化和未来。

开发LSTMs，使RNNs能够学习长期依赖关系。发现香草RNNs不能很好地处理相关事件(通常被称为教师信号)的工作时间超过了几分钟。香草RNNs受爆炸/消失梯度问题的困扰，该问题阻碍了模型学习收敛到最优参数。LSTMs通过引入一个内存单元来消除这个问题，它可以确保不断的错误流和控制单元来控制进出单元的信息流。原始的LSTM体系结构在输入数据中的序列有很好的定义边界时工作得很好，但是失败了。补救办法是忘记盖茨，他会根据需要自动刷新LSTM内存。从那时起，已经提出了一些变化。突出的一个是窥视孔连接，使LSTMs能够学习精确的间隔。

然而，所有这些复杂的因素都是以增加复杂性为代价的。人们普遍认为，LSTMs很难进行训练和优化，因为需要调整许多不同的参数。通过使用已经训练过的模型和已知的参数值，可以在实践中获得良好的结果，但是需要大量的努力来理解LSTMs的工作原理以及它们的局限性。像[21], [46]这样的研究是为了了解各种LSTM组件的目的，并分析LSTM的不同变体是否确实提供了任何优势。在[47]中提出了一种新的简化体系结构，称为门控循环单元(GRU)，并已相当流行。这是另一个迹象，表明标准的LSTM可能会非常强大，我们很可能在将来看到更简单的变体。

46

第六章

结论

本文利用LSTM网络对时态数据进行了异常检测。我们通过研究LSTMs来解释其复杂的架构、学习远程依赖的能力以及维护LSTM状态的不同方法。在此基础上，提出了一种基于综合预测模型的异常检测方法。我们简要介绍了异常检测研究中普遍存在的各种问题。为了规避这些问题，我们选择了三个现实世界的数据集，这些数据集已经在以前的研究中使用过，并且包含了不同类型的异常。最后，我们对这些数据集进行了实验。基于我们的研究结果，我们得出结论:LSTMs是有效的时间序列模型和异常检测器。但我们建议先尝试前馈，因为它们在很多情况下都是足够的。LSTMs具有额外的复杂性，只有在数据具有远程依赖和多个模式时才会提供优势。

6.1未来的工作

在[2]和[38]中讨论了异常检测中的一些挑战，包括数据集的选择和算法的评估。虽然这些建议都不是针对时间序列的，但我们相信尝试它们仍然是有益的。我们使用的异常检测方法直接用于点异常。为探测集体异常，可以开发利用滑动窗口中预测误差之和的方法。我们还希望使用更复杂的数据集，在多个时间尺度上包含远程依赖和模式，这样可以更显著地展示LSTMs的优势。最后，在简化LSTM架构的过程中，尝试GRUs是很有趣的。GRUs将输入合并到一个更新门中，并没有保留一个显式的内存单元。由于这个简化的架构，GRUs在计算上比LSTMs更有效。他们也被发现在LSTMs[48]中产生了类似的结果，并且正逐渐成为一种流行的替代方法。

47

参考书目

[1]Chandola, A. Banerjee，和V. Kumar，“异常检测:一项调查”ACM Comput。测量员第41卷。。3、pp. 15:1-15:58, 2009年7月。doi:

10.1145/1541880.1541882。(在线)。:http://doi.acm.org/10.1145/ 1541880.1541882

[2]G. O. Campos, A. Zimek, J. Sander, R. J. g.b. Campello, B. Micenkova, E. Schubert, I. Assent, M. E. Houle。

非监督离群检测:测量、数据集和实证研究，“数据挖掘和知识发现，第30卷，第30期，第30期。2016年7月1日doi:10.1007 / s10618 - 015 - 0444 - 8。(在线)。可用:http://dx.doi.org/10.1007/s10618 - 015 - 0444 - 8

[3]杨和吴，《数据挖掘研究中的10个挑战》，《国际信息技术期刊》;决策，vol. 05，不。04年,

597 - 604年,597页。doi:10.1142 / S0219622006002258。(在线)。可用:http://www.worldscientific.com/doi/abs/10.1142/S0219622006002258

[4]. Gupta, J. Gao, C. C. Aggarwal，和J. Han，“对时间数据的异常检测:一项调查，”IEEE的知识和数据工程的交易，

26卷,没有。9、2014年9月2250-2267页。doi:10.1109 / TKDE.2013.184

[5]. LeCun, Y. Bengio, G. Hinton，《深度学习》，《自然》，第521卷，不。7553年,第444 - 436页,436年。doi:10.1038 / nature14539。(在线)。可用:

http://dx.doi.org/10.1038/nature14539

[6]A. Hakansson，“研究项目和学位项目的研究方法和方法的门户”，国际会议论文集。

关于教育的前沿:计算机科学与计算机工程fecs13，第67-73页，2013年。(在线)。可用:http://urn.kb.se/resolve?瓮= urn:nbn公司禁止:se:k:女主角- 136960

[7]《人工神经网络导论》第1部分。“(在线)。可用:http://www.theprojectspot.com/tutorial-post/

introduction-to-artificial-neural-networks-part-1/7

[8]D. Williams和G. Hinton，“通过反向传播错误的学习表现”，Nature, vol. 323, no。6088年,第538 - 533页,533年。

48

参考书目

[9]S. Ruder，“梯度下降优化算法概述”，arXiv预印arXiv:1609.04747, 2016。

[10]I. Goodfellow, Y. Bengio and A. Courville, Deep Learning。麻省理工学院出版社,2016年,http://www.deeplearningbook.org。

[11]P. Domingos，“关于机器学习的一些有用的知识”，ACM的通讯，第55卷，不。10日,78 - 87年,78页。

[12]R. J. Frank, N. Davey，和S. P. Hunt，《时间序列预测与神经网络》，《智能&机器人系统杂志》，第31卷，第31卷。1、91 - 91页。

2001年。

[13]联合。“使用深度信念网络的声学建模”，IEEE在音频、语音和语言支持方面的交易。

国际消费电子展上,20卷,不。1,14-22,2012页。

[14]L. Rabiner and B. Juang，“对隐马尔科夫模型的介绍”，IEEE ASSP杂志，第3卷，第3期。1986年1月4-16日。doi:

10.1109 / MASSP.1986.1165342

[15]P. J. Werbos，“通过时间的反向传播:它做什么和如何做”，IEEE, vol. 78，不。10日,1550 - 1560年,1550页。

[16]. Bengio, P. Simard和P. Frasconi，“学习长期依赖梯度下降是困难的，”IEEE在神经网络上的交易，

5卷,没有。2、157 - 166年,157页。

[17]S. Hochreiter, Y. Bengio, P. Frasconi, J. Schmidhuber，“复发网的梯度流:学习长期依赖的困难”，2001。

(在线)。可用:http://www.bioinf.jku.at/publications/older/ch7.pdf

[18]R. Pascanu, T. Mikolov和Y. Bengio，“关于训练复发性神经网络的困难”。“ICML (3)， vol. 28, pp. 1310-1318, 2013。

[19]S. Hochreiter和J. Schmidhuber，“长期短时记忆”，神经计算，第9卷，第9期，第9期。8,1735 - 1780年,1735页。

[20]F. A. Gers, J. Schmidhuber和F. Cummins，“学会遗忘:LSTM的持续预测，”神经计算，第12卷，第12期，第12期。10页2451 - 2471

2000年。

[21]K. Greff, R. K. Srivastava, J. KoutnAk, B. R. Steunebrink, J. schmeshuber，“LSTM:搜索空间奥德赛”，IEEE在Neu上的交易。

拉尔网络和学习系统，vol. PP，不。99年1 - 11,2017页。doi:10.1109 / TNNLS.2016.2582924

49

参考书目

[22]M. Hermans和B. Schrauwen，“训练和分析深层递归神经网络”，在神经信息处理系统的研究进展。坏蛋,

ran Associates, Inc.， 2013, pp. 190-198。(在线)。可用:http://papers.nips。cc /纸/ 5166 -培训-和-分析-深- -神经- networks.pdf复发

[23]格雷夫斯，A. r.默罕默德，G.辛顿，“深度复发神经网络的语音识别”，2013年IEEE国际会议。

2013年5月，关于声学、语音和信号处理的研究。doi:10.1109 / ICASSP.2013.6638947。ISSN 1520 - 6149 6645 - 6645页。

[24]H. Sak, A. Senior，和F. Beaufays，“长短期记忆重复神经网络结构，用于大规模声学建模”。

2014年国际语言交流协会第十五届年会。(在线)。可用:http://193.6.4.39/ ~ czap / letoltes / IS14 / IS2014 / PDF /作者/ IS141304.PDF

[25]. Basu和M. Meckesheimer，《时间序列的自动离群检测:传感器数据的应用》，《知识与信息系统》，第11卷，

不。2、137 - 154年,137页。doi:10.1007 / s10115 - 006 - 0026 - 6。(在线)。可用:http://dx.doi.org/10.1007/s10115 - 006 - 0026 - 6

[26]d.j.希尔和B. S. Minsker，“流环境传感器数据的异常检测:数据驱动建模方法”，环境建模。

软件，vol. 25，不。9日,第1022 - 1014页,1014年。

[27]J. Ma和S. Perkins，《关于时间序列的在线新颖性检测》，第九届ACM SIGKDD国际会议论文集。

知识发现和数据挖掘，ser。KDD ' 03。纽约，纽约，美国:ACM, 2003。doi:10.1145/956750.956828。ISBN 1-58113-737-0页613 - 618。(在线)。可用:http://doi.acm.org/10.1145/956750.956828

[28]R. S. Tsay, D. Pena和A. E. Pankratz，“多变量时间序列中的异常值”，生物统计学，第87卷，第87期。4、789 - 804年,789页。

[29]Y. Yao, A. Sharma, L. Golubchik，和R. Govindan，“对传感器系统的在线异常检测:一种简单而有效的方法，”性能。

评价,卷67,不。11日,第1075 - 1059页,1059年。

[30]H. N. Akouemo和R. J. Povinelli，《天然气时间序列数据的概率异常检测》，《国际预测期刊》，第32卷，第32期。3,

948 - 956年,948页。

[31]Malhotra, L. Vig, G. Shroff，和P. Agarwal，“时间序列异常检测的长期短期记忆网络”。

鲁汶大学出版社，2015年，第89页。(在线)。可用:https://www.elen.ucl.ac.be/proceedings/esann/esannpdf/es2015 - 56. - pdf

50

参考书目

[32]S. Chauhan和L. Vig，“在ECG时间信号中的异常检测通过长期的短期记忆网络”，2015年IEEE国际Con-。

数据科学与高级分析(DSAA)， 2015年10月。doi:10.1109 / DSAA.2015.7344872 1 - 7页。

[33]L. Bontemps, V. L. Cao, J. McDermott和n.a.。一种基于长短时记忆递归神经网络的集体异常检测方法。

的工作原理。施普林格国际出版，2016，第141-152页。ISBN 978-3-319-48057-2。(在线)。可用:http://dx.doi.org/10.1007/978 - 3 - 319 - 48057 - 2 - \_9

[34]程先生、徐志强、刘丽华、刘小东、李启超、王永强，“m -LSTM: BGP异常检测的多尺度LSTM模型”，在网络Proto-中。

cols (ICNP)， 2016 IEEE 24国际会议。IEEE,2016年,页1 - 6。

[35]. Marchi, F. Vesperini, F. Weninger, F. Eyben, S. Squartini, and B. Schuller，“用LSTM回归神经网络进行声学非线性预测”。

2015年7月神经网络国际联席会议(IJCNN)。doi:10.1109 / IJCNN.2015.7280757。ISSN 2161 - 4393 1 - 7页。

[36]Malhotra, A. Ramakrishnan, G. Anand, L. Vig, P. Agarwal和G. Shroff，基于lstm的多传感器异常编码解码器。

检测，" CoRR, vol. abs/1607.00148, 2016。(在线)。可用:http:/ /arxiv.org/abs/1607.00148

[37]斯诺克，H.拉罗谢尔，R. P. Adams，“机器学习算法的实用贝叶斯优化”，在Neu的研究中。

ral信息处理系统25。柯兰联合公司，2012年，第2951-2959页。(在线)。可用:http://papers.nips。cc /纸/ 4522 -实践-贝叶斯-优化-机-学习- algorithms.pdf

[38]F. Emmott, S. Das, T. Dietterich, A. Fern, and w - k。“系统构建异常检测基准”。

数据，“在ACM SIGKDD研讨会的会议上关于离群检测和描述，ser。奇怪的13。纽约，纽约，美国:ACM, 2013。doi:10.1145/2500853.2500858。ISBN 978-1-4503-2335-2 16页。(在线)。可用:http://doi.acm.org/10.1145/2500853.2500858

[39]A. Lavin和S. Ahmad，“评估实时异常检测算法——Numenta异常基准”，2015年IEEE第14国际。

机器学习与应用会议(ICMLA)， 2015年12月。doi:10.1109 / ICMLA.2015.141 38-44页。

[40]E. Keogh, J. Lin, A. Fu，“热SAX:高效地找到最不寻常的时间序列”，第五届IEEE国际数据会议。

51

参考书目

矿业(ICDM ' 05),2005年11月。doi:10.1109 / ICDM.2005.79。ISSN 1550 - 4786 8 -页。

[41]琼斯，D. Nikovski, M. Imamura，和T. Hirata，“在实值多维时间序列中的异常检测”，在国际会议上。

Bigdata / Socialcom /网络安全。斯坦福大学,ASE 2014。(在线)。可用:http://www.merl.com/publications/docs/tr2014 - 042. - pdf

[42]M. C. Chuah and F. Fu, ECG异常检测通过时间序列分析。柏林，海德堡:柏林海德堡，2007年，第123-135页。

ISBN 978-3-540-74767-3。(在线)。可用:http://dx.doi.org/10.1007/ 978 - 3 - 540 - 74767 - 3 - \_14

[43]J.霍金斯和D.乔治，“分层的时间记忆:概念，理论和术语，”技术报告，Numenta, Tech. Rep.， 2006。(在线)。

可用:http://www - edlab.cs.umass.edu/cs691jj/hawkins -和-乔治- 2006。pdf

[44]. Mikolov， "基于神经网络的统计语言模型"，博士论文，Brno科技大学。,2012年。(在线)。可用:

http://www.fit.vutbr.cz/ imikolov / rnnlm / thesis.pdf

[45]F. A. Gers, D. Eck和J. Schmidhuber，通过时间窗方法将LSTM应用于时间序列。柏林,海德堡:施普林格

柏林海德堡，2001年，第69 - 676页。ISBN 978-3-540-44668-2。(在线)。可用:http://dx.doi.org/10.1007/3 - 540 - 44668 - 0 - \_93

[46]R. Jozefowicz, W. Zaremba，和I. Sutskever，“对经常性网络架构的实证探索”，在第32届国际会议的会议上。

机器学习会议(ICML-15)， 2015，第2342-2350页。

[47]K. Cho, B. Van Merrienboer, C. Gulcehre, D. Bahdanau, F. Bougares, H. Schwenk，和Y. Bengio。

《统计机器翻译的RNN编码解码器》，《arXiv预印arXiv:1406.1078, 2014》。(在线)。可用:https://arxiv.org/abs/1406.1078

[48]J. Chung, C. Gulcehre, K. Cho，和Y. Bengio，“基于序列模型的门控递归神经网络的实证研究”，arXiv预印本。

arXiv:1412.3555,2014。(在线)。可用:https://arxiv.org/abs/1412.3555

52