

论文阅读笔记

论文阅读笔记

Anomaly Detection for Time Series Based on the Neural Networks Optimized by the Improved PSO Algorithm

相关知识

PSO

LSTM

CNN

论文摘要&简介

IPSO-CLSTM: The Proposed Approach for Time Series Anomaly Detection

整体框架

时间序列预测: CLSTM Optimized by the Improved PSO

异常检测器: Anomaly Detector

对比实验

数据集

评价指标

实验结果

Anomaly Detection for Time Series Based on the Neural Networks Optimized by the Improved PSO Algorithm

https://link.springer.com/chapter/10.1007/978-3-030-60796-8_20

相关知识

PSO

粒子群优化算法(PSO: Particle swarm optimization) 是一种进化计算技术 (evolutionary computation)。源于对鸟群捕食的行为研究。

基本思想

它的基本思想是: 粒子群算法通过设计一种无质量的粒子来模拟鸟群中的鸟, 粒子仅具有两个属性: **速度和位置**, 速度代表移动的快慢, 位置代表移动的方向。每个粒子在搜索空间中单独的搜寻最优解, 并将其记为当前个体极值, 并将个体极值与整个粒子群里的其他粒子共享, 找到最优的那个个体极值作为整个粒子群的当前全局最优解, 粒子群中的所有粒子根据自己找到的当前个体极值和整个粒子群共享的当前全局最优解来调整自己的速度和位置。

更新规则

PSO初始化为一群随机粒子(随机解)。然后通过迭代找到最优解。在每一次的迭代中，粒子通过跟踪两个“极值”(pbest, gbest)来更新自己。在找到这两个最优值后，粒子通过下面的公式来更新自己的速度和位置。

公式解读：

公式 (1) :

$$v_i = v_i + c_1 \times rand() \times (pbest_i - x_i) + c_2 \times rand() \times (gbest_i - x_i)$$

公式 (2) :

$$x_i = x_i + v_i$$

在公式 (1)、(2) 中， $i=1, 2, \dots, N$, N 是此群中粒子的总数。

v_i : 是粒子的速度

$rand()$: 介于 (0, 1) 之间的随机数

x_i : 粒子的当前位置

c_1 和 c_2 : 是学习因子，通常 $c_1 = c_2 = 2$

v_i 的最大值为 V_{\max} (大于0)，如果 v_i 大于 V_{\max} ，则 $v_i = V_{\max}$

公式 (1)、(2) 为 PSO 的 **标准形式**。 [csdn.net/daaikuaichuan](https://www.csdn.net/daaikuaichuan)

$$\text{公式 (1)} : v_i = v_i + c_1 * rand() * (pbest_i - x_i) + c_2 * rand() * (gbest_i - x_i)$$

公式 (1) 的第一部分称为「**记忆项**」，表示上次速度大小和方向的影响；公式 (1) 的第二部分称为「**自身认知项**」，是从当前指向粒子自身最好点的一个矢量，表示例子的动作来源于自己经验的部分；公式 (1) 的第三部分成为「**群体认知项**」，是一个从当前点指向种群最好点的矢量，反映了粒子间的协同合作和知识共享。

粒子就是通过自己的经验和同伴中最好的经验来决定下一步的运动。以上面两个公式为基础，形成了 **PSO 的标准形式**。

公式 (3) :

$$v_i = \omega \times v_i + c_1 \times rand() \times (pbest_i - x_i) + c_2 \times rand() \times (gbest_i - x_i)$$

ω 叫做惯性因子，其值为非负。

其值较大，全局寻优能力强，局部寻优能力弱；

其值较小，全局寻优能力弱，局部寻优能力强。

动态 ω 能获得比固定值更好的寻优结果。动态 ω 可在 PSO 搜索过程中线性变化，也可以根据 PSO 性能的某个测度函数动态改变。

目前采用较多的是线性递减权值 (Linearly Decreasing Weight, LDW) 策略。

$$\omega^{(t)} = (\omega_{ini} - \omega_{end})(G_k - g) / G_k + \omega_{end}$$

G_k : 最大迭代次数

ω_{ini} : 初始惯性权值

ω_{end} : 迭代至最大进化代数时的惯性权值

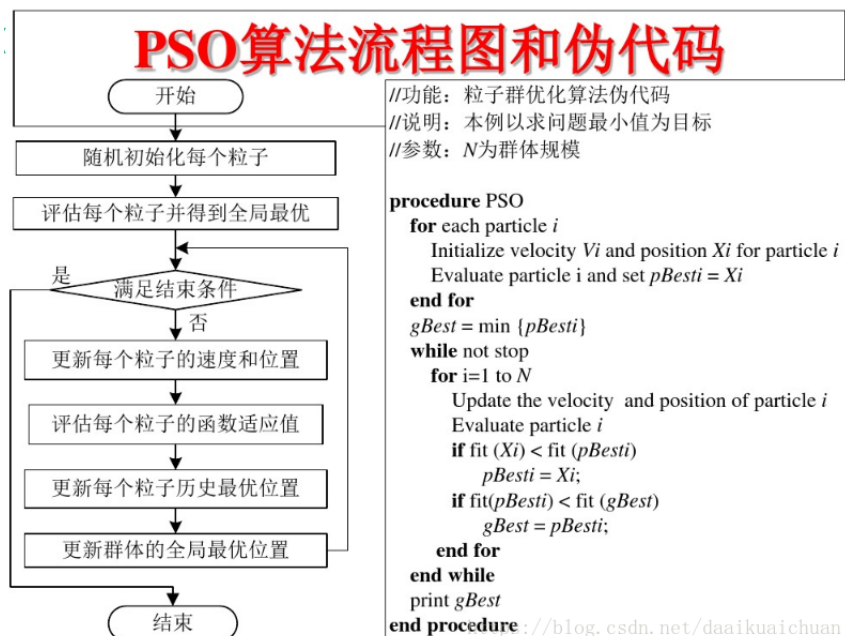
典型权值:

$$\omega_{ini}=0.9, \omega_{end}=0.4$$

ω 的引入，使用 PSO 算法性能有了很大的提高，针对不同的搜索问题，

可以调整全局和局部搜索能力，也使 PSO 算法有成功地应用于很多实际问题。 [csdn.net/daaikuaichuan](https://www.csdn.net/daaikuaichuan)

PSO 算法的流程和伪代码



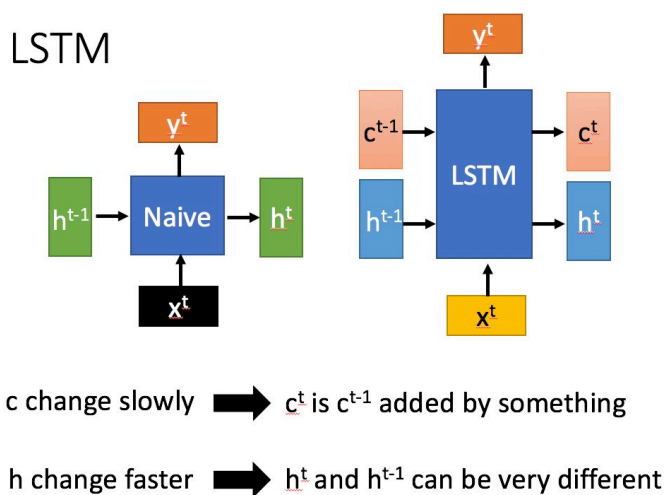
具体示例：<https://blog.csdn.net/daaikuaichuan/article/details/81382794>

优点

PSO的优势：在于简单容易实现并且没有许多参数的调节。目前已被广泛应用于函数优化、神经网络训练、模糊系统控制以及其他遗传算法的应用领域。

LSTM

<https://zhuanlan.zhihu.com/p/32085405>



相比RNN只有一个传递状态 h^t ，LSTM有两个传输状态，一个 c^t （cell state），和一个 h^t （hidden state）。（Tips：RNN中的 h^t 对于LSTM中的 c^t ）其中对于传递下去的 c^t 改变得很慢，通常输出的 c^t 是上一个状态传过来的 c^{t-1} 加上一些数值。而 h^t 则在不同节点下往往会有很大的区别。

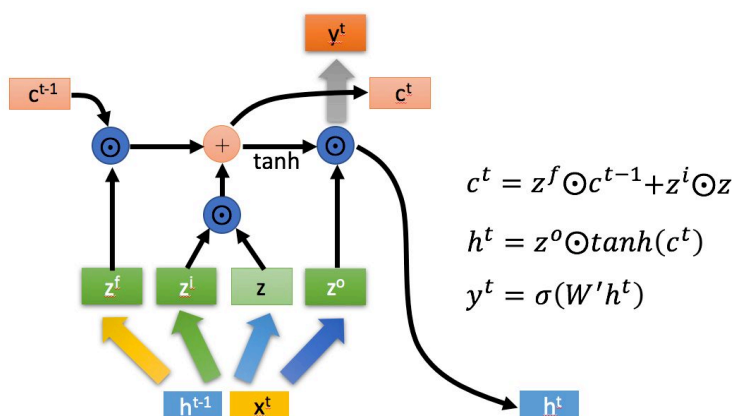
下面具体对LSTM的内部结构来进行剖析。

首先使用LSTM的当前输入 x^t 和上一个状态传递下来的 h^{t-1} 拼接训练得到四个状态。

$$z = \tanh\left(W \begin{bmatrix} x^t \\ h^{t-1} \end{bmatrix}\right)$$

$$z^i = \sigma\left(W^i \begin{bmatrix} x^t \\ h^{t-1} \end{bmatrix}\right)$$

其中， z^f ， z^i ， z^o 是由拼接向量乘以权重矩阵之后，再通过一个 **sigmoid** 激活函数转换成0到1之间的数值，来作为一种门控状态。而 z 则是将结果通过一个 **tanh** 激活函数将转换成-1到1之间的值（这里使用 **tanh** 是因为这里是将其做为输入数据，而不是门控信号）。



⊙ 是Hadamard Product，也就是操作矩阵中对应的元素相乘，因此要求两个相乘矩阵是同型的。⊕ 则代表进行矩阵加法。

LSTM内部主要有三个阶段：

\1. **忘记阶段**。这个阶段主要是对上一个节点传进来的输入进行**选择性**忘记。简单来说就是会“忘记不重要的，记住重要的”。

具体来说是通过计算得到的 z^f （f表示forget）来作为忘记门控，来控制上一个状态的 c^{t-1} 哪些需要留哪些需要忘。

\2. **选择记忆阶段**。这个阶段将这个阶段的输入有选择性地“记忆”。主要是会对输入 x^t 进行选择记忆。哪些重要则着重记录下来，哪些不重要，则少记一些。当前的输入内容由前面计算得到的 z 表示。而选择的门控信号则是由 z^i （i代表information）来进行控制。

将上面两步得到的结果相加，即可得到传输给下一个状态的 c^t 。也就是上图中的第一个公式。

\3. **输出阶段**。这个阶段将决定哪些将会被当成当前状态的输出。主要是通过 z^o 来进行控制的。并且还上一阶段得到的 c^o 进行了放缩（通过一个tanh激活函数进行变化）。

与普通RNN类似，输出 y^t 往往最终也是通过 h^t 变化得到。

CNN

CNN is a type of artificial neural network that has been widely used in computer vision (CV) and natural language processing (NLP). As the name indicates, the network employs a convolution operation. Normally, CNN consists of **convolutional layers, pooling layers, and fully connected layers**.

论文摘要&简介

当前，有关时间序列的异常检测方法研究的工作已经做了很多了，比较常见的是利用ARIMA模型进行异常检测，但是它的局限在于当数据中存在明显的趋势或自相关时，ARIMA模型才相对有效，但是在现实生活中数据往往是比较复杂且受到多方面的影响的，因此现在比较流行使用异常检测方法是先建立预测模型，利用预测值与真值之间的差异来计算异常评分检测异常。深度网络例如LSTM和CNN已经成功运用到了时间序列的预测，但是他们在时间序列异常检测的运用却不常见，而且这些算法这些算法的性能在很大程度上取决于它们的超参数值。因此，寻找到一个能够获得最优参数值的有效方法是十分重要的。这篇论文中，他们设想了一个新的架构将改进的PSO和CNN+LSTM结合起来，提出一种改进的粒子群算法IPSO（improved particle swarm optimization）引入了一种非线性递减分配方法来提高惯性权重，从而实现了关键参数的自动搜索，优化时间序列异常检测模型的超参数。论文还与其他进化算法相比，显示了IPSO与CLSTM的结合具有良好的性能。

IPSO-CLSTM: The Proposed Approach for Time Series Anomaly Detection

整体框架

IPSO-CLSTM分为两个模块：1、时间序列预测模块，负责预测给定了范围的时间戳；2、第二个模块“异常检测器”负责将给定时间序列数据点标记为正常还是异常。

具体如下所示：

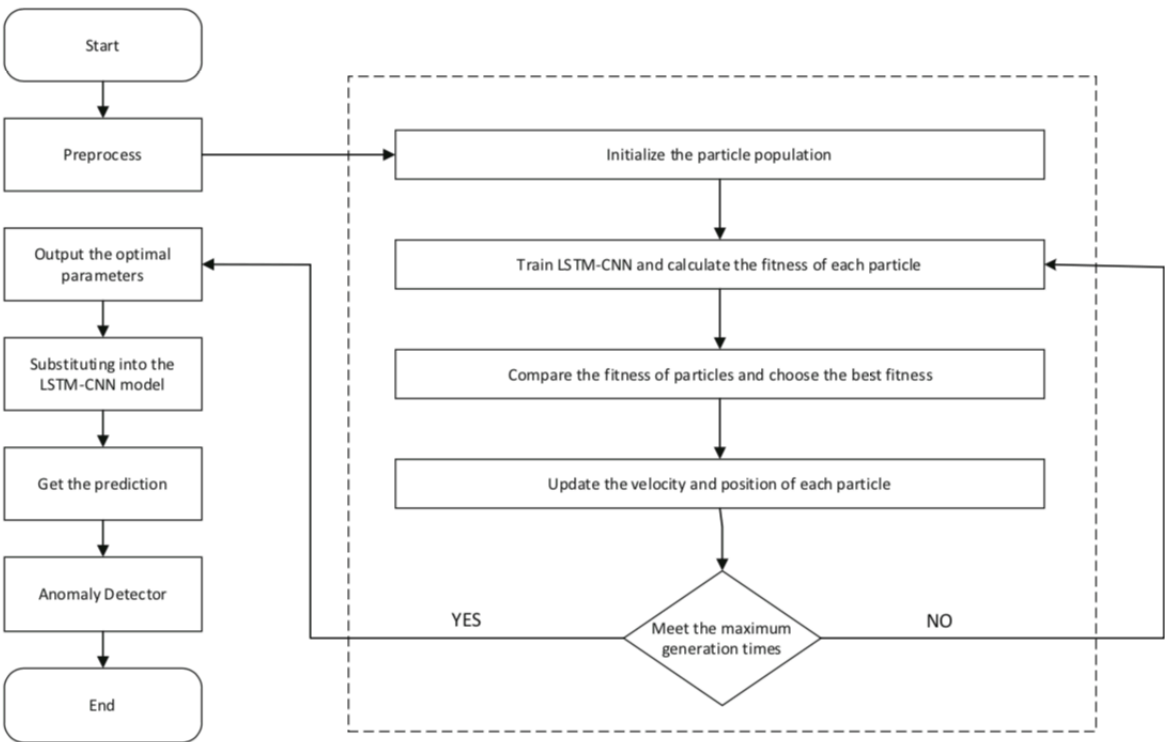


Fig. 2. Architecture summary of the proposed approach

时间序列预测：CLSTM Optimized by the Improved PSO

LSTM能够对时间序列进行很好的预测，但是LSTM缺乏提取本地上下文信息的能力。为了提高性能，并由于CNN的参数效率，本文将1D Conv（一维卷积）运用到了论文中。此外，由于CNN-LSTM的重要参数时间窗口、LSTM和1D CNN的cell number等的不同，获得最佳解决方案或接近最佳解决方案的性能表现也不同。在本文中，使用了单个隐藏层LSTM，一个卷积层和一个完全连接层。

PSO的改进

PSO是一种典型的进化算法，惯性权重 w 用于控制迁移速度对当前粒子速度的影响，表现为PSO的性能。传统的惯性权重分配是线性分配，随着迭代次数线性减少，PSO的局部搜索能力会变差。本文采用非线性减量分配方法。

$$w = w_{max} - (w_{max} - w_{min}) * \sqrt{\frac{i}{iter_{max}}}$$

其中 w_{max} 和 w_{min} 分别代表最大和最小惯性权重， i 是当前迭代次数， $iter_{max}$ 是最大迭代次数。

在PSO优化中，**粒子被初始化为（look_back, node1, node2）**。Look_back代表时间窗口，node1是LSTM中具有单个隐藏层的隐藏层中包含的节点数，node2是1D Conv中包含的节点数。在本文中，将训练数据输入到神经网络中进行训练，使用Adam训练LSTM的内部参数。在每次迭代时，系统都会为每个粒子计算速度和新位置。选择**训练数据的均方根误差（RMSE）作为个体适应度函数**，并将**最小适应度值作为改进PSO的迭代目标**。改进的PSO用于查找要优化的最佳参数，以确定最佳预测模型。

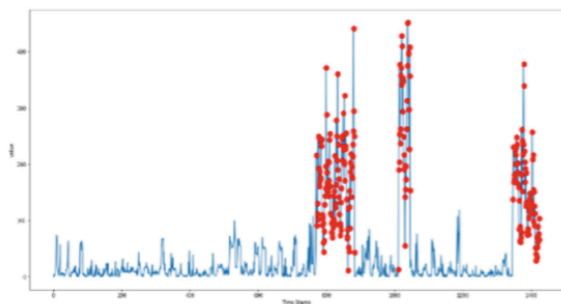
异常检测器：Anomaly Detector

一旦时间序列预测模块给出了下一个时间戳 $x_t + 1$ 的预测。该模型执行检测异常的功能。计算出实际值与预测值之差。在新数据上，将计算错误的对数概率密度（PDFs），并将其用作异常评分，值越小，表示观察到的异常可能性越大。在此给出阈值，以确定大多数异常检测算法所需的时间戳异常或正常。

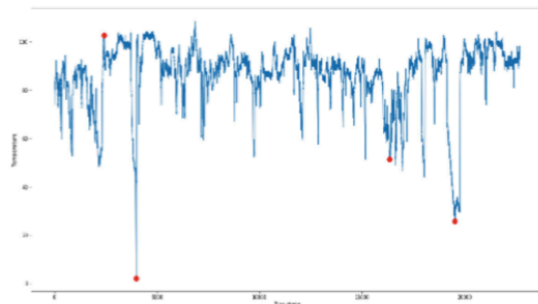
对比实验

数据集

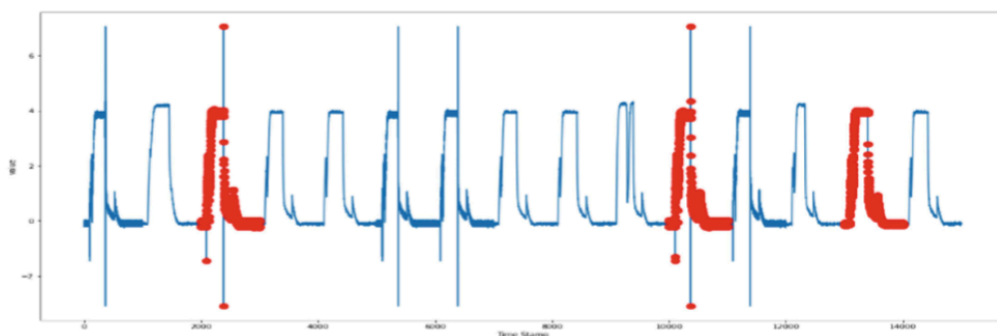
- YAHOO webscope S5 datasets
- Numenta Machine Temperature Dataset
- NASA Space Shuttle Valve Dataset



(a) Yahoo Webscope S5



(b) NAB Temperature



(c) NASA space shuttle valve data

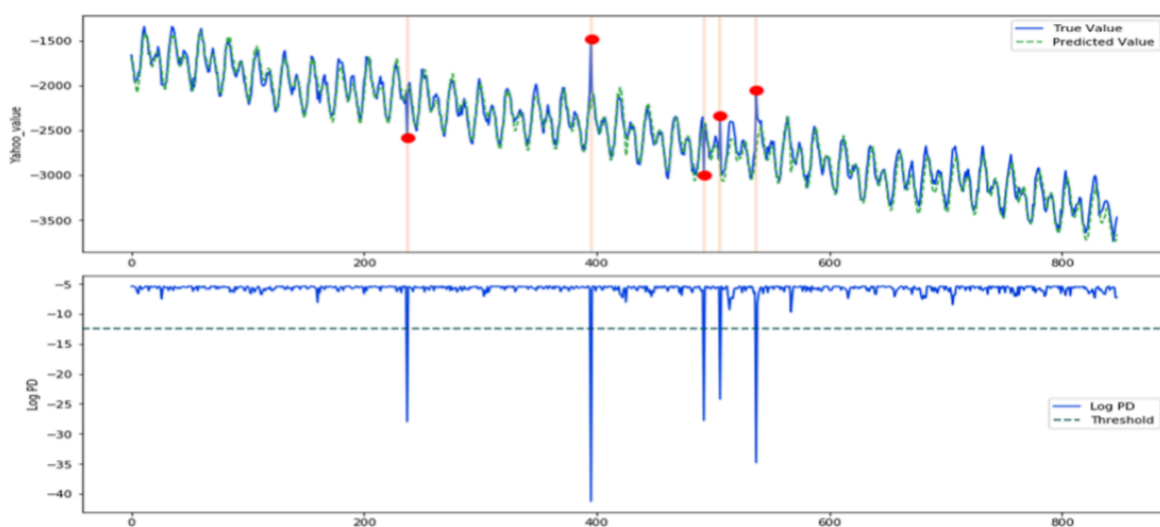
评价指标

In this paper, various models, including the proposed model are evaluated using **precision**, **recall**, and **F-score**.

实验结果

a. YAHOO webscope S5 datasets

我们发现，与整个Yahoo基准测试中的其他算法相比，所提出的算法具有更好的性能。



如图所示的适应度值的变化表明改进的PSO具有更快的收敛速度。

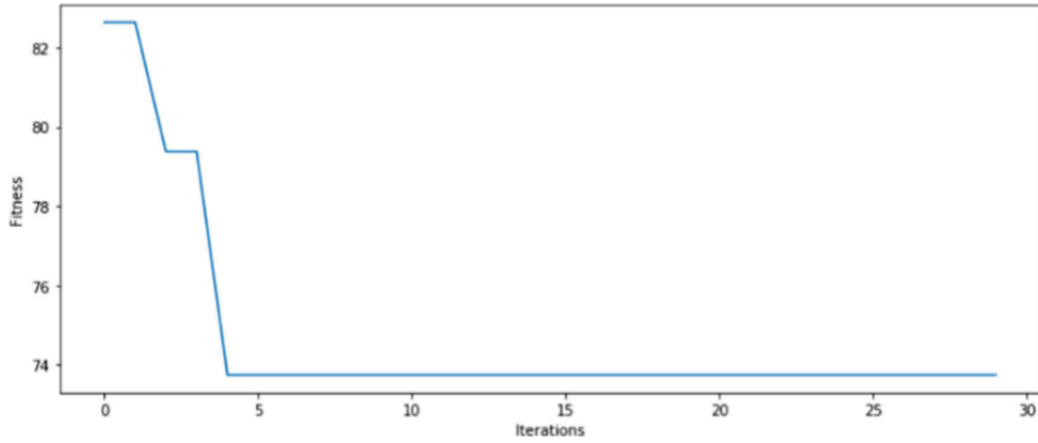


Fig. 4. The change of fitness value

b. Numenta Machine Temperature Dataset

在这个数据集上，论文提出的模型表现更好。

ps: 论文说“It may be noted here that IPSO-CLSTM outperforms other algorithms in **recall which is more important to** anomaly detection than **precision in the industrial scene.**”

Table 2. Performance comparison of the proposed models with other methods

Method	ContextOSE [11]	Numenta [12]	Skyline [13]	IPSO-CLSTM
Recall	0.001	0.004	0.01	1
F-score	0.002	0.008	0.020	0.014

c. NASA Space Shuttle Valve Dataset

在前面的实验中，我们已经证明了IPSO-CLSTM具有检测时间序列数据中点异常的能力。在本节中，我们表明IPSO-CLSTM也适用于异常子序列（Time series discords）检测。异常子序列是一个较长时间序列的子序列，它不同于子序列的其余部分。在该数据集中，每个子序列由1000个数据点组成，下面的实验结果表明，IPSO-CLSTM也适用于检测异常子序列。

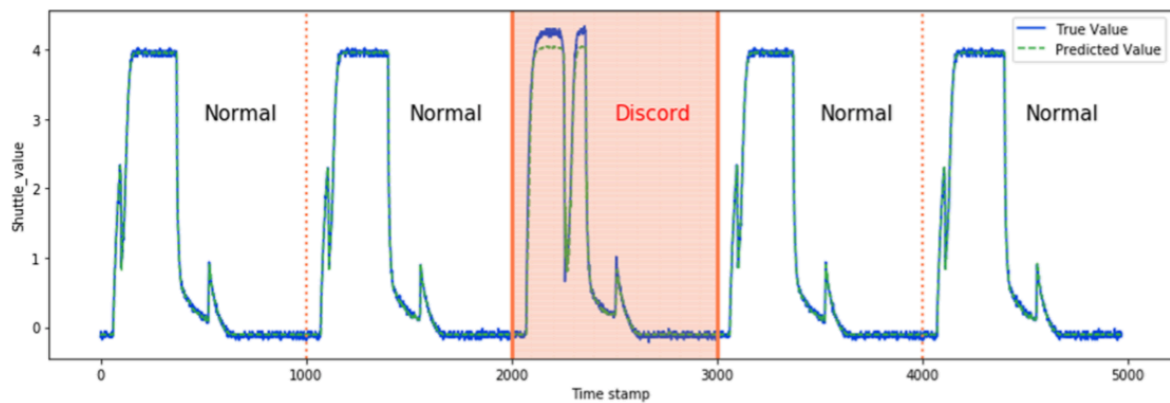


Fig. 7. Shaded in peach denotes detections made by the IPSO-CLSTM.