(19)中华人民共和国国家知识产权局

(12)发明专利申请

(10)申请公布号 CN 112966714 A
(43)申请公布日 2021.06.15

(21)申请号 202110142428.X

(22)申请日 2021.02.02

(71)申请人 湖南大学
地址 410081 湖南省长沙市岳麓区麓山路36号

(72)发明人 吴迪 戴宁一 邓晗晖 江中凯 谢小峰 范喆 聂祥

(74)专利代理机构 长沙新裕知识产权代理有限公司 43210
代理人 梁小林

(51)Int.Cl.
G06K 9/62(2006.01)
G06N 3/04(2006.01)

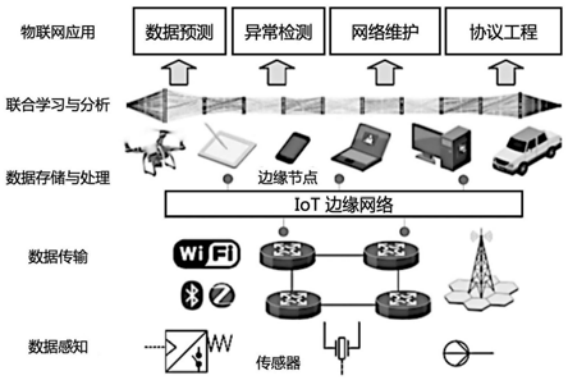
权利要求书3页 说明书6页 附图4页

(54)发明名称

一种边缘时序数据异常检测和网络可编程控制方法

(57)摘要

本发明涉及一种边缘时序数据异常检测和网络可编程控制方法,属于物联网时序数据与深度学习以及机器学习结合的领域。通过获取物联网边缘设备上的时序数据;根据基于Grid LSTM的注意力机制对物联网时序数据进行预测;通过基于Grid LSTM的注意力机制预测模型对边缘设备上的物联网时序数据进行预测得到真实值与预测值之间的误差;利用SVM算法来对上述误差进行异常检测,得到数据的异常情况;实现异常数据包的传输路径的追溯和屏蔽,以及数据新的传输路径的查找。本发明的有益效果在于,提高了对物联网时序数据的分析与处理的能力,数据预测性能和异常检测性能都提高了;解决了无线传感器网络数据传输时的数据安全问题。



CN 112966714 A

1.一种边缘时序数据异常检测和网络可编程控制方法,其特征在于:

提出了一种边缘长短时记忆网络系统的概念,即EdgeLSTM,一种将深度学习中注意力机制、Grid LSTM和机器学习算法中SVM相结合的思想;

通过获取物联网边缘设备上的时序数据;

根据基于Grid LSTM的注意力机制对物联网时序数据进行预测;

通过基于Grid LSTM的注意力机制预测模型对边缘设备上的物联网时序数据进行预测得到真实值与预测值之间的误差;

利用SVM算法来对上述误差进行异常检测,得到数据的异常情况;从而提出一种网络可编程控制方法,实现异常数据包的传输路径的追溯和屏蔽,以及数据新的传输路径的查找。

2.根据权利要求1所述的一种边缘时序数据异常检测和网络可编程控制方法,其特征在于,具体包括以下步骤:

步骤一、对原始数据进行探索性分析,

对于采集到的原始传感器的数据,进行初步的数据探索分析,查看一般变量与一般变量之间的相关性,以及一般变量与目标变量之间的相关性;观察每个变量的缺失值、异常值情况;

步骤二、数据的预处理,

原始数据集是利用多个传感器采集到的数据,因此需要对原始数据集进行预处理,主要包括数据清洗、数据填充、数据下采样以及使用$x_{std} = \frac{x-min}{\max - min}$对数据进行归一化,其中min和max分别表示某一列特征的值的最小值和最大值,x表示该特征的所有值,x_{std}表示归一化后的值,取值范围为[0,1];

步骤三、数据集的划分,

将经过预处理之后的数据集进行分割,按照6:2:2的比例分割成训练集、验证集和测试集,其中训练集只包含了正常的数据,而测试集和验证集中既包含了正常的数据也包含了异常的数据;

步骤四、搭建基于Enhanced Grid LSTM的注意力预测模型,

物联网时序数据中异常值的数据量比较小,使用训练集来对预测模型进行搭建,通过验证集为预测模型选择超参数以达到更好的效果,使用公式如下:

$$g^u = \sigma(W^u H)$$

$$g^f = \sigma(W^f H)$$

$$g^o = \sigma(W^o H)$$

$$g^c = \tanh(W^c H) \qquad (1)$$

$$m' = g^f \otimes m + g^u \otimes g^c$$

$$h' = \tanh(g^o \otimes m')$$

其中，σ是逻辑sigmoid函数，其表达式为$sigmoid(x)=\frac{1}{1+e^{-x}}$，$W^u$，$W^f$，$W^o$，$W^c$分别表示不同状态下的权重矩阵；$H=[I*x_i,h]^T$，其中$x_i$表示当前的输入，I表示转换后的映射矩阵，h表示前一时刻的输出向量；g^u表示输入门，用来决定将要更新的信息；g^f表示遗忘门，用来决定需要丢弃什么信息；g^o表示输出门，用来决定将要输出的信息到下一个细胞状态中；g^c表示当前将要更新的信息到新的细胞当中；m′表示当前时刻记忆单元状态的输出，h′表示当前时刻隐藏单元状态的输出；

根据上述最基本LSTM神经网络的框架，通过这个框架将N维隐藏向量$h_1,h_2,\cdots,h_i,\cdots,h_N$和N维记忆向量$m_1,m_2,\cdots,m_i,\cdots,m_N$作为输入参数，最后输出N维隐藏向量$h'_1,h'_2,\cdots,h'_i,\cdots,h'_N$和N维记忆向量$m'_1,m'_2,\cdots,m'_i,\cdots,m'_N$，具体公式如下所示：

$$(h'_1,m'_1)=\mathrm{LSTM}(H,m_1,W_1)$$

$$\vdots$$

$$(h'_i,m'_i)=\mathrm{LSTM}(H,m_i,W_i) \quad (2)$$

$$\vdots$$

$$(h'_N,m'_N)=\mathrm{LSTM}(H,m_N,W_N)$$

其中W_i(i＝1,2,…,N)是权重矩阵W_i^u，W_i^f，W_i^o，W_i^c拼接而成的权重矩阵；对于每一个单元格，网格有N个边来接收隐藏状态向量和记忆状态向量，并且输出N个隐藏状态向量和记忆状态向量，一个数据点沿着某一侧的一对输入隐藏/记忆状态向量映射到Grid LSTM网络中；在边缘长短时记忆网络(EdgeLSTM)系统中，使用2维Grid LSTM单元，h_1和h_2分别表示时间维度和深度维度上的隐藏向量，m_1和m_2分别表示时间和深度方向上的记忆向量；因此在时间维度上使用h_1和m_1来进行2D网格LSTM单元计算，最后输出的是隐藏状态向量h'_1和记忆状态向量m'_1；相应地，在深度维度上对h_2和m_2进行计算，并得到隐藏状态向量h'_2和记忆状态向量m'_2；h_1和h_2产生了上述方程式1中使用的各种门控机制，并且m_1和m_2被组合成学习物联网时序数据复杂特征的主要记忆状态向量；在构建2D网格LSTM单元后，将这些单元连接起来形成2D网格LSTM网络，是由四个单元通过循环连接组成的，水平轴代表时间维度，垂直轴代表深度维度；步骤五、数据的预测，

利用测试集来对已训练好的预测模型进行测试并评估，最后我们会得到正常数据的预测值和异常数据的预测值；由于是机器学习中的回归问题，采用的评估指标是平均绝对百分比误差(Mean Absolute Percentage Error，MAPE)、均方根误差(Root Mean Square Error，RMSE)，平均绝对值误差(Mean Absolute Error，MAE)和R^2分数，具体计算公式如下：

$$MAPE=\frac{1}{n}\sum_{i=1}^{n}\frac{|y_t^i-\hat{y}_t^i|}{y_t^i} \quad (3)$$

$$RMSE=\sqrt{\frac{1}{m}\sum_{i=1}^{m}(y_t^i-\hat{y}_t^i)^2} \quad (4)$$

$$MAE = \frac{1}{m}\sum_{i=1}^{m}|y_t^i - \hat{y}_t^i| \quad (5)$$

$$R^2 = 1 - \frac{\sum_{i=1}^{n}(y_t^i - \hat{y}_t^i)^2}{\sum_{i=1}^{n}(y_t^i - \bar{y}_t)^2} \quad (6)$$

其中，y_t^i表示t时刻第i个样本的真实值，$\hat{y}_t^i$表示t时刻第i个样本的预测值，n表示样本的总数，$\bar{y}_t$表示t时刻样本的均值。MAPE的值越小越好，最小值是0；R^2的值越大越好，最大值为1，表示模型对未知数据的拟合效果最好；

步骤六、数据异常检测，

通过步骤四、五，获得验证集在预测模型上的预测值以及测试集在预测模型上的预测值，并分别根据获得的预测值来构建验证集残差数据集和测试集残差数据集，然后使用验证集残差数据集来构建多类SVM检测模型，使用测试集残差数据集来测试多类SVM异常检测模型；采用的分类评估标准是精准率：Precision、召回率：Recall、F_β分数，具体计算公式如下：

$$Precision = \frac{TP}{TP + FP} \quad (7)$$

$$Recall = \frac{TP}{TP + FN} \quad (8)$$

$$F_\beta = \frac{(1+\beta^2) * Precision * Recall}{\beta^2 * Precision + Recall} \quad (9)$$

其中TP、TN、FP和FN是分类模型输出的四种结果，TP表示将正类预测为正类的数目，TN表示将负类预测为负类的数目，FP表示将负类预测为正类的数目，FN表示将正类预测为负类的数目；

步骤七、网络的可编程控制

在边缘服务器对所接收的数据进行异常检测后，若为异常数据，则通过接入点发起关于该数据包的追溯查询，接入点会通过广播方式发送一个包含异常数据包的源节点ID和异常数据包的ID的数据包；其周围所有节点从该数据包中提取出源节点的ID和异常数据包的ID，将它们以公式(10)的方式进行拼接，并查询是否在当前传感器节点的布鲁姆过滤器中存储；

Hash(pId||sId||lId||hId) (10)

其中，Hash表示其中一个哈希函数，pId表示当前数据包的ID，sId表示发送该数据包的源节点的ID，lId表示当前节点的ID，nId表示下一跳节点的ID，||表示拼接运算；经过循环迭代，直至最终节点与异常数据包的源节点的ID相同为止；为了避免这个可疑节点，源节点会通过新的路径进行数据包的发送，使源节点通过新路径传输数据。

一种边缘时序数据异常检测和网络可编程控制方法

技术领域

[0001] 本发明涉及一种边缘时序数据异常检测和网络可编程控制方法，属于物联网时序数据与深度学习以及机器学习结合的领域。

背景技术

[0002] 随着智慧城市，工业4.0，供应链以及家庭自动化技术的发展，物联网(Internet of Things,IoT)应用已经产生了大量数据。根据思科的报告，到2021年，连接的物联网设备数量将达到116亿，意味着每月将产生超过49艾字节的数据流量。无处不在的传感器产生了大量的数据与信息，而这些数据正在成为物联网计算中最普遍的数据形式，使得数据传输和处理在物联网应用中起着越来越关键的作用。通过对物联网数据的处理可以获得非常有用和有价值的信息，以便为这些物联网应用程序进行智能自动化和决策提供保障。在受数据驱动的物联网应用对延迟，可靠性、安全性以及实时性都有严格的要求，由于受到网络带宽和计算资源的限制，物联网需要预加载流量，因此我们需要部署一个平台，该平台在网络边缘集成了连接、计算、存储等功能，而边缘计算的出现则满足了这些需求，因此边缘计算已成为在边缘执行数据处理和智能化的一种解决方案。网络，即最接近物联网数据的源头。同时，由于物联网数据是由分布式智能设备和传感器生成的，因此物联网数据具有大规模的特点，不同的数据采集设备使得物联网中采集到的数据类型是多种多样的；由于位于特定位置的设备采集到的传感器数据标有时间戳，使得物联网数据之间具有相互的依赖性这也是物联网数据与传统数据最显著的差异，物联网数据对实时性要求很高，例如，当某传感器设被运行异常的时候，则需要立即检测，以避免影响其他设备的正常运行。因此，需要一种有效且高效的系统来分析和处理边缘的物联网时间序列数据。当前对于时序数据和时空数据的预测虽然在实际生活中有较多的应用场景，但对于无线传感器网络大多是对节点的节能分析使用，很少使用这些方法考虑无线传感器节点的安全性问题。因此，需要对时序数据和时空数据的预测提出了在无线传感器网络中新的应用场景，以解决无线传感器网络在数据传输过程中数据被篡改的不安全问题，实现了无线传感器网络数据传输过程中的网络自动维护，确保了无线传感器能够安全传输数据。

发明内容

[0003] 本发明的目的在于提供一种边缘时序数据异常检测与预测分析方法，从而克服现有技术中的不足。

[0004] 本发明的技术方案在于，提出了一种边缘长短时记忆网络(EdgeLSTM)系统的概念，具体是指一种将深度学习中注意力机制、Grid LSTM与机器学习算法中SVM相结合的思想。利用EdegLSTM系统，提取到物联网时序数据的多维特征，并在网络边缘进行灵活和稳定的处理。边缘长短时记忆网络(EdgeLSTM)的关键思想是利用注意力机制来提取多维特征的重要性，将LSTM单元扩展为网格结构，并利用Grid LSTM深度计算物联网时序数据。Grid LSTM沿着任意维度部署单元。在边缘长短时记忆网络(EdgeLSTM)系统中，沿着时间(水平)

方向和深度(垂直)方向进行单元格的部署,与标准的LSTM网络相比,边缘长短时记忆网络(EdgeLSTM)可以处理具有多维并且特征更加复杂的数据的处理。具体来说,边缘长短时记忆网络(EdgeLSTM)系统使用基于Grid LSTM神经网络的注意力机制来预测时间序列数据的趋势,然后使用多类支持向量机(Multiclass SVM)来进行异常检测的分类。本发明的边缘长短时记忆网络(EdgeLSTM)系统可以充分发挥边缘计算的潜力,通过物联网边缘数据驱动处理改善网络系统的管理。将边缘长短时记忆网络(EdgeLSTM)部署到三个数据驱动物联网应用程序上,即数据预测、异常检测和网络维护。并且通过一种网络可编程控制方法,实现异常数据包的传输路径的追溯和屏蔽,以及数据新的传输路径的查找,从而处理无线传感器网络数据传输过程中的安全问题,维护传感器节点。

[0005] 本发明的有益效果在于,提高了对物联网时序数据的分析与处理的能力,同时对物联网时序数据进行异常检测;与不具有边缘长短时记忆网络(EdgeLSTM)的模型相比,数据预测性能和异常检测性能都提高了。所提出的网络可编程控制方法,提高了数据传输网络的可靠性和安全性。

附图说明

[0006] 图1为本发明的一种在边缘网络中物联网计算架构图。

[0007] 图2为本发明的一种面向物联网时序数据处理与分析的NeuroIoT框架图。

[0008] 图3为本发明的Grid LSTM单元结构图。

[0009] 图4为本发明的Grid LSTM网络结构图。

[0010] 图5为本发明基于EnhancedGrid LSTM的注意力预测模型结构图。

[0011] 图6为本发明在电力数据集上的预测结果图。

[0012] 图7为本发明在SML2010数据集上的预测结果图。

[0013] 图8为异常数据包的路径追溯图

具体实施方式

[0014] 为了更加清楚地说明本发明实施例中的技术方案,下面结合附图作1-8作进一步描述;显然,所描述的实施例仅仅为本发明中的一部分实施例,并不是全部的实施例。

[0015] 图1为本发明实施例公开的一种在边缘网络中物联网计算架构图,该图主要包括数据感知层、数据传输层、数据的存储和处理层、联合学习与分析层以及物联网应用层。

[0016] 图2为本发明实施例公开的面向物联网时序数据处理与分析的NeuroIoT框架图,该方法包括:

步骤一、对原始数据进行探索性分析

具体的,在本实例中,对于采集到的原始传感器的数据,本发明进行初步的数据探索分析,主要是查看一般变量与一般变量之间的相关性,以及一般变量与目标变量之间的相关性;观察每个变量的缺失值、异常值情况。

[0017] 步骤二、数据的预处理

具体的,在本实例中,原始数据集是利用多个传感器采集到的数据,因此需要对原始数据集进行预处理,主要包括数据清洗、数据填充、数据下采样以及使用$x_{std} = \frac{x-min}{\max-min}$

对数据进行归一化，其中min和max分别表示某一列特征的值的最小值和最大值，x表示该特征的所有值，x_{std}表示归一化后的值，取值范围为[0,1]。

[0018] 步骤三、数据集的划分

具体的，在本实例中，将经过预处理之后的数据集进行分割，按照6:2:2的比例分割成训练集、验证集和测试集，其中训练集只包含了正常的数据，而测试集和验证集中既包含了正常的数据也包含了异常的数据。

[0019] 步骤四、搭建基于Enhanced Grid LSTM的注意力预测模型

具体的，在本实例中，由于物联网时序数据中异常值的数据量比较小，所以使用训练集来对预测模型进行搭建，然后通过验证集来为预测模型选择超参数以达到更好的效果，本发明的预测模型结构图如图5所示。对于模型搭建过程中，使用了一些公式具体如下：

$$g^u=\sigma(W^uH)$$
$$g^f=\sigma(W^fH)$$
$$g^o=\sigma(W^oH)$$
$$g^c=\tanh(W^cH) \quad (1)$$
$$m'=g^f\otimes m+g^u\otimes g^c$$
$$h'=\tanh(g^o\otimes m')$$

[0020] 其中，σ是逻辑sigmoid函数，其表达式为$sigmoid(x)=\frac{1}{1+e^{-x}}$，$W^u$，$W^f$，$W^o$，$W^c$分别表示不同状态下的权重矩阵。$H=[I*x_i,h]^T$，其中$x_i$表示当前的输入，I表示转换后的映射矩阵，h表示前一时刻的输出向量。g^u表示输入门，用来决定什么信息将要更新；g^f表示遗忘门，用来决定需要丢弃什么信息；g^o表示输出门，用来决定将要输出什么信息到下一个细胞状态中；g^c表示当前将要更新什么信息到新的细胞当中；m′表示当前时刻记忆单元状态的输出，h′表示当前时刻隐藏单元状态的输出。上述是一个最基本LSTM神经网络的框架，通过这个框架将N维隐藏向量$h_1,h_2,\cdots,h_i,\cdots,h_N$和N维记忆向量$m_1,m_2,\cdots,m_i,\cdots,m_N$作为输入参数，最后输出N维隐藏向量$h'_1,h'_2,\cdots,h'_i,\cdots,h'_N$和N维记忆向量$m'_1,m'_2,\cdots,m'_i,\cdots,m'_N$，具体公式如下所示：

$$(h'_1,m'_1)=\mathrm{LSTM}(H,m_1,W_1)$$
$$\vdots$$
$$(h'_i,m'_i)=\mathrm{LSTM}(H,m_i,W_i) \quad (2)$$
$$\vdots$$
$$(h'_N,m'_N)=\mathrm{LSTM}(H,m_N,W_N)$$

其中W_i(i=1,2,…,N)是权重矩阵W_i^u,W_i^f,W_i^o,W_i^c拼接而成的权重矩阵。

[0021] Grid LSTM网络结构图如图3所示：对于每一个单元格，网格有N个边来接收隐藏状态向量和记忆状态向量，并且输出N个隐藏状态向量和记忆状态向量，一个数据点沿着某一侧的一对输入隐藏/记忆状态向量映射到Grid LSTM网络中。在边缘长短时记忆网络(EdgeLSTM)系统中，使用2维Grid LSTM单元，其结构图如图3所示，h_1和h_2分别表示时间(水

平)维度和深度(垂直)维度上的隐藏向量,m_1和m_2分别表示时间和深度方向上的记忆向量。因此在时间维度上使用h_1和m_1来进行2D网格LSTM单元计算,最后输出的是隐藏状态向量h'_1和记忆状态向量m'_1;相应地,在深度维度上对h_2和m_2进行计算,并得到隐藏状态向量h'_2和记忆状态向量m'_2。h_1和h_2产生了上述方程式1中使用的各种门控机制,并且m_1和m_2被组合成学习物联网时序数据复杂特征的主要记忆状态向量。在构建2D网格LSTM单元后,将这些单元连接起来形成2D网格LSTM网络,如下图4所示,它是由四个单元通过循环连接组成的,水平轴代表时间维度,垂直轴代表深度维度。具体来讲,时间维度中的输入对应于时间序列,并且每个隐藏层的单元格对应于不同的时间步。对于当前时刻的输出,它将不同时刻的数据考虑在内,并评估它们在下一个时刻的影响。对于每个Grid LSTM单元,它通过上述介绍的门控机制来控制数据的输入、存储以及输出。当前单元的门控机制接收之前隐藏层(例如h_1^{t-1})生成的前一时刻的输出。输入当前时刻样本,然后确定当前隐藏层的输出(例如h_1^t),这将在下一个时刻用于生成下一个隐藏层的输出(例如h_1^{t+1})。对于沿着深度维度进行信息处理,它的工作流程与时间维度类似,当前单元的门控机制处理来自前一个隐藏层(例如h_2^{l-1})的时间序列,并生成当前隐藏层的输出(例如h_2^l),相应的导出下一个隐藏层的(例如h_2^{l+1})的输出。在边缘长短时记忆网络(EdgeLSTM)中,将时间维度中的网格LSTM单元作为深度维度的输入,然后将深度维度中的网格LSTM单元作为时间维度的输入。通过与注意力机制思想结合起来,更好的提取时序数据重要性高的特征。通过执行此替代过程,最终对来自时间和深度维度的序列数据进行建模并进行预测。与单个LSTM网络相比,的边缘长短时记忆网络(EdgeLSTM)可以增强模型学习物联网数据中更加复杂特征的能力。

[0022] 步骤五、数据的预测

具体的,在本实例中,利用测试集来对已训练好的预测模型进行测试并评估,最后会得到正常数据的预测值和异常数据的预测值。由于是机器学习中的回归问题,所以采用的评估指标是平均绝对百分比误差(Mean Absolute Percentage Error,MAPE)、均方根误差(Root Mean Square Error,RMSE),平均绝对值误差(Mean Absolute Error,MAE)和R^2分数,具体计算公式如下:

$$MAPE = \frac{1}{n}\sum_{i=1}^{n}\frac{|y_t^i - \hat{y}_t^i|}{y_t^i} \quad (3)$$

$$RMSE = \sqrt{\frac{1}{m}\sum_{i=1}^{m}(y_t^i - \hat{y}_t^i)^2} \quad (4)$$

$$MAE = \frac{1}{m}\sum_{i=1}^{m}|y_t^i - \hat{y}_t^i| \quad (5)$$

$$R^2 = 1 - \frac{\sum_{i=1}^{n}(y_t^i - \hat{y}_t^i)^2}{\sum_{i=1}^{n}(y_t^i - \bar{y}_t)^2} \quad (6)$$

其中,y_t^i表示t时刻第i个样本的真实值,$\hat{y}_t^i$表示t时刻第i个样本的预测值,n表示

样本的总数，$\bar{y}_t$表示t时刻样本的均值。MAPE的值越小越好，最小值是0；R^2的值越大越好，最大值为1，表示模型对未知数据的拟合效果最好。图6和图7所示，在两个测试数据集上的预测结果，同时表1，也显示了本发明的预测模型相对于其它模型来说，效果最佳。

Model	SML 2010 dataset				Power dataset			
	RMSE	MAE	MAPE (%)	R^2	RMSE	MAE	MAPE (%)	R^2
LSTM	0.0459	0.0378	0.1660	0.974	0.5940	0.4731	4.115	0.942
BiLSTM	0.0386	0.0323	0.1512	0.980	0.5620	0.4381	3.959	0.951
GRU	0.0422	0.0356	0.1535	0.979	0.6039	0.5210	4.376	0.937
MLP	0.0531	0.0472	0.2310	0.952	0.7150	0.6360	6.176	0.861
Attention-LSTM	0.0341	0.0223	0.1229	0.988	0.4670	0.3092	3.745	0.960
Enhanced Grid LSTM	**0.0273**	**0.0194**	**0.0917**	**0.999**	**0.3820**	**0.2968**	**3.481**	**0.972**

表1不同模型在两个数据集上的预测性能对比

[0023] 步骤六、数据异常检测

具体的，在本实例中，通过步骤四、五，可以获得验证集在预测模型上的预测值以及测试集在预测模型上的预测值，并分别根据获得的预测值来构建验证集残差数据集和测试集残差数据集，然后使用验证集残差数据集来构建多类SVM检测模型，使用测试集残差数据集来测试多类SVM异常检测模型。在这里采用的分类评估标准是精准率(Precision)、召回率(Recall)、F_β分数，具体计算公式如下：

$$Precision = \frac{TP}{TP + FP} \tag{7}$$

$$Recall = \frac{TP}{TP + FN} \tag{8}$$

$$F_\beta = \frac{(1 + \beta^2) * Precision * Recall}{\beta^2 * Precision + Recall} \tag{9}$$

其中，TP、TN、FP和FN是分类模型输出的四种结果，TP表示将正类预测为正类的数目，TN表示将负类预测为负类的数目，FP表示将负类预测为正类的数目，FN表示将正类预测为负类的数目。F分数是精准率和召回率之间的权衡，是模型性能的综合考虑，F分数越高，说明本文的模型综合性能越好。一般通过设置β大小来决定在模型评估时更偏向于哪个指标，当β＜1时，精准率的权重大于召回率；当β＞1时，精准率的权重小于召回率；当β＝1时，精准率的权重和召回率的权重一样大。当做检索时，一般要求先提升召回率，然后再提升精准率，也就是召回率的权重大于精准率；当做疾病检测、异常检测的时候，一般要求先提升精准率，再提升召回率，也就是精准率的权重大于召回率。在本文实验中，由于是做异常检测任务，所以认为精准率比召回率更加重要，因此，设置β＝0.1。结果如表2所示，可以看出，本发明的异常检测模型相对于其它模型，效果较好。

表2不同模型在两个数据集上的分类性能对比

Method	Power dataset			SML2010 dataset		
	Precision	Recall	F_β-score	Precision	Recall	F_β-score
MLP	0.723	0.854	0.776	0.816	0.892	0.852
GRU	0.859	0.894	0.876	0.869	0.901	0.884
LSTM-AutoEncoder	0.885	0.912	0.898	0.873	0.917	0.894
LSTM-Guassian	0.892	0.925	0.908	0.884	0.929	0.906
BiLSTM-AutoEncoder	0.873	0.895	0.883	0.902	0.938	0.920
GridLSTM	0.904	0.933	0.918	0.910	0.941	0.925
NeuroIoT	**0.921**	**0.947**	**0.934**	**0.932**	**0.958**	**0.948**

[0024] 步骤七、网络的可编程控制

当数据进行传输时，在各个节点中都设定一个布鲁姆过滤器进行数据的存储。当数据包经过传感器节点时，该节点将数据包的识别信息即数据包的编号ID、发送该数据包的源节点的ID、本地节点的ID以及下一跳路由节点的ID拼接形成新的字符串，然后将该字符串通过哈希映射放入该节点的布鲁姆过滤器中，并将相应位设为1。其实现如公式(10)所示：

$$Hash(pId||sId||lId||nId) \tag{10}$$

其中，Hash表示其中一个哈希函数，pId表示当前数据包的ID，sId表示发送该数据包的源节点的ID，lId表示当前节点的ID，nId表示下一跳节点的ID，||表示拼接运算。

[0025] 在边缘服务器对所接收的数据进行异常检测后，若为异常数据，则通过接入点发起关于该数据包的追溯查询。接入点会通过广播方式发送一个包含异常数据包的源节点ID和异常数据包的ID的数据包。由于广播的特性，其周围所有节点都会接收到该数据包，它们从该数据包中提取出源节点的ID和异常数据包的ID，然后将它们以公式(10)的方式进行拼接，并查询是否在当前传感器节点的布鲁姆过滤器中存储。

[0026] 如果在当前布鲁姆过滤器中，则说明该异常数据包可能经过该节点，则该节点继续上述操作即发送相应广播数据包，并屏蔽该节点，使得该节点无法进行数据的传输。否则，不进行任何操作。

[0027] 其具体追溯过程如图8所示，当边缘服务器由于检测到来自于节点1的数据异常，通过接入点发起对该数据包的追溯请求。接入点会通过广播的方式发送含有所需信息的查询数据包，而在其广播范围内的节点6、节点8和节点9会收到该查询数据包，会提取该数据包的信息，进行拼接并查询异常数据包是否通过该节点。若节点6的布鲁姆过滤器命中，则认为该节点是该异常数据包的发送路径上的节点。节点6会屏蔽自身节点，同时继续以广播的方式发送查询数据包给其邻居节点3、节点5、节点7和接入节点，而接入节点已经判定处于异常数据包的路径上，因此不会再次发送查询数据包，其余邻居节点则会继续进行上述操作，直至其到达节点1为止。而后，节点1会重新查找新的传输路径传输数据至接入点。

[0028] 如此循环迭代，直至最终节点与异常数据包的源节点的ID相同为止。因为处于路径上的节点都有可能被攻击，为了避免这个可疑节点，源节点会通过新的路径进行数据包的发送。

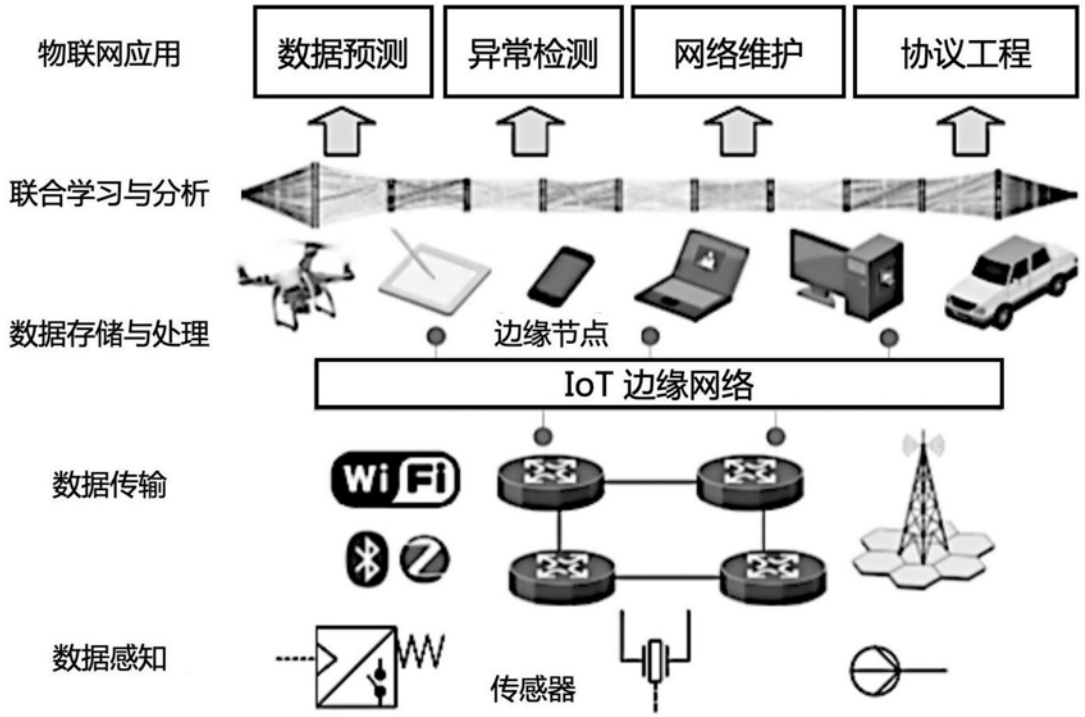

物联网应用
数据预测
异常检测
网络维护
协议工程
联合学习与分析
数据存储与处理
边缘节点
IoT 边缘网络
数据传输
WiFi
数据感知
传感器

图1

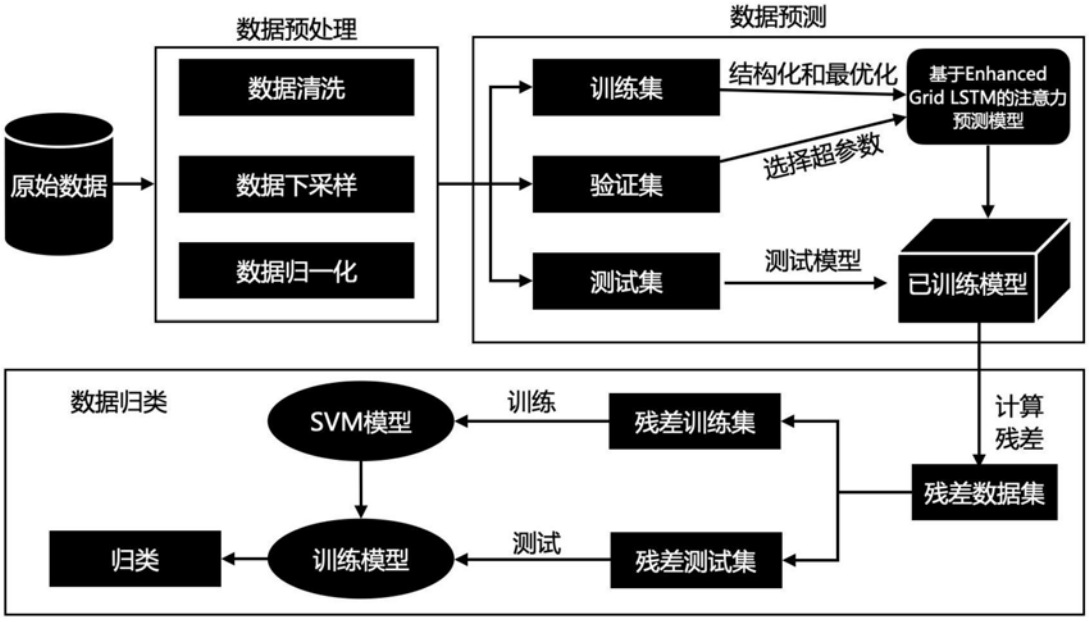

数据预处理
数据预测
数据清洗
数据下采样
数据归一化
原始数据
训练集
验证集
测试集
结构化和最优化
选择超参数
基于Enhanced Grid LSTM的注意力预测模型
测试模型
已训练模型
数据归类
SVM模型
训练
残差训练集
计算残差
残差数据集
归类
训练模型
测试
残差测试集

图2

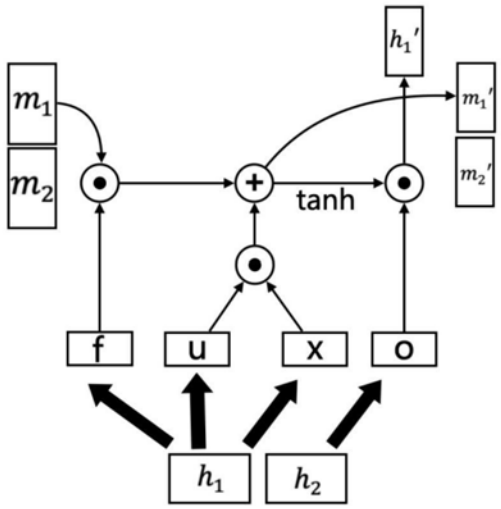

m_1
m_2
h_1'
m_1'
m_2'
tanh
f
u
x
o
h_1
h_2

图3

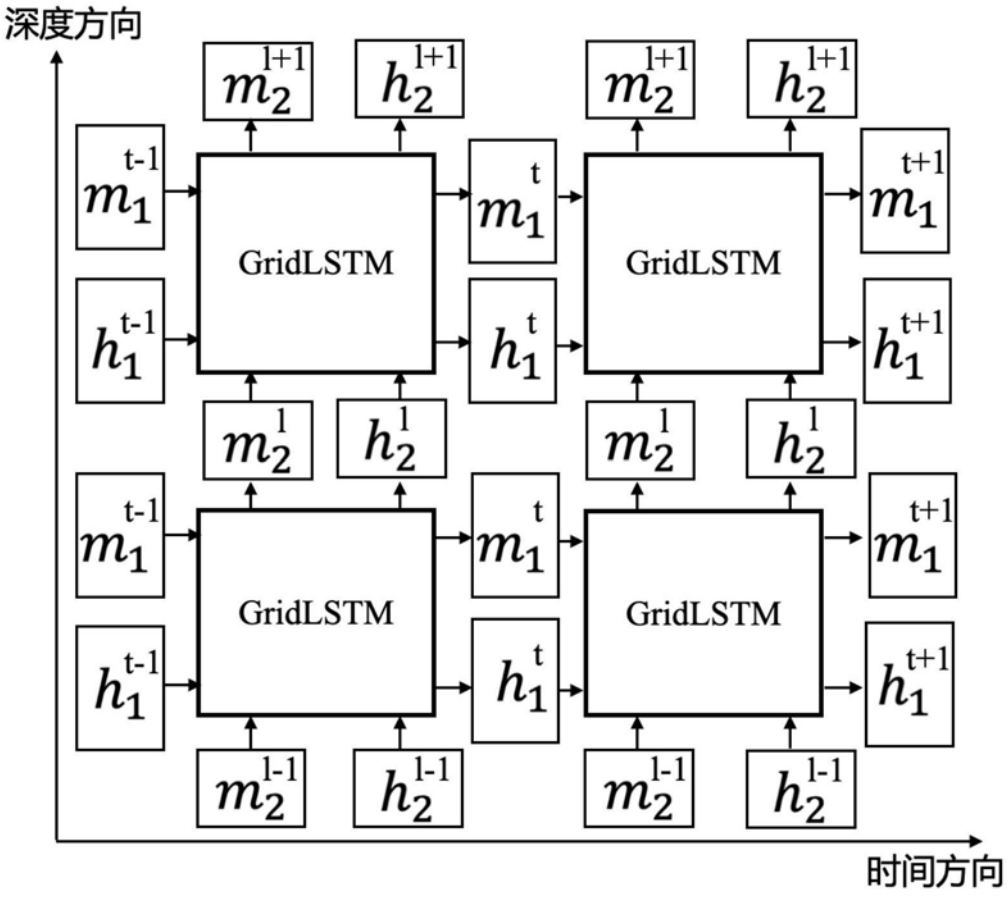

深度方向
GridLSTM
GridLSTM
GridLSTM
GridLSTM
时间方向

图4

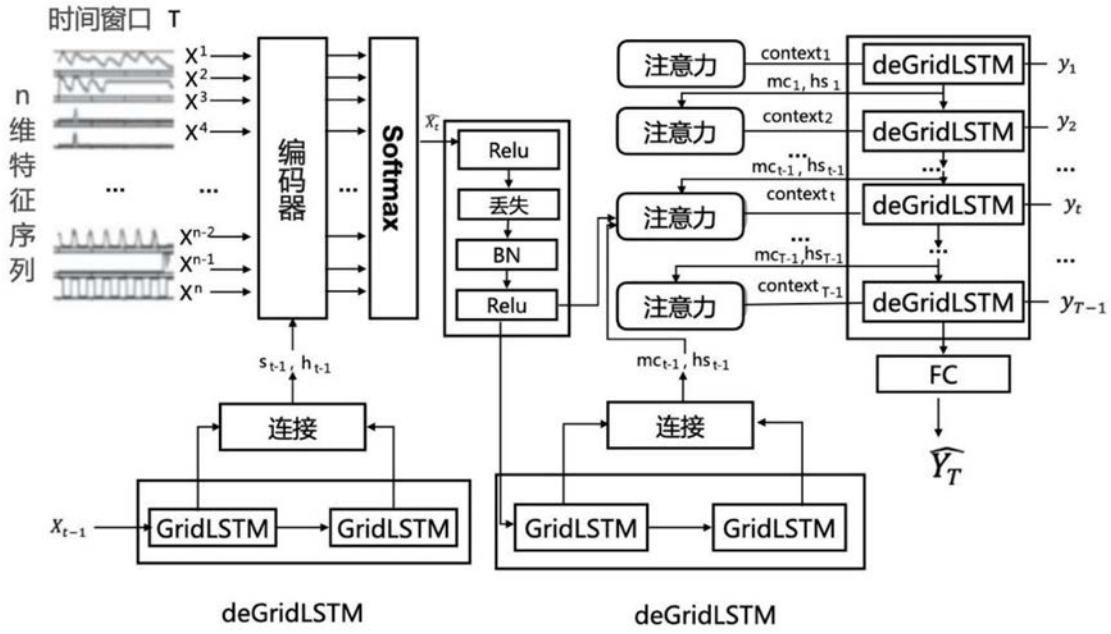

时间窗口 T
n维特征序列
编码器
Softmax
Relu
丢失
BN
Relu
注意力
context1
deGridLSTM
FC
连接
GridLSTM
deGridLSTM
deGridLSTM

图5

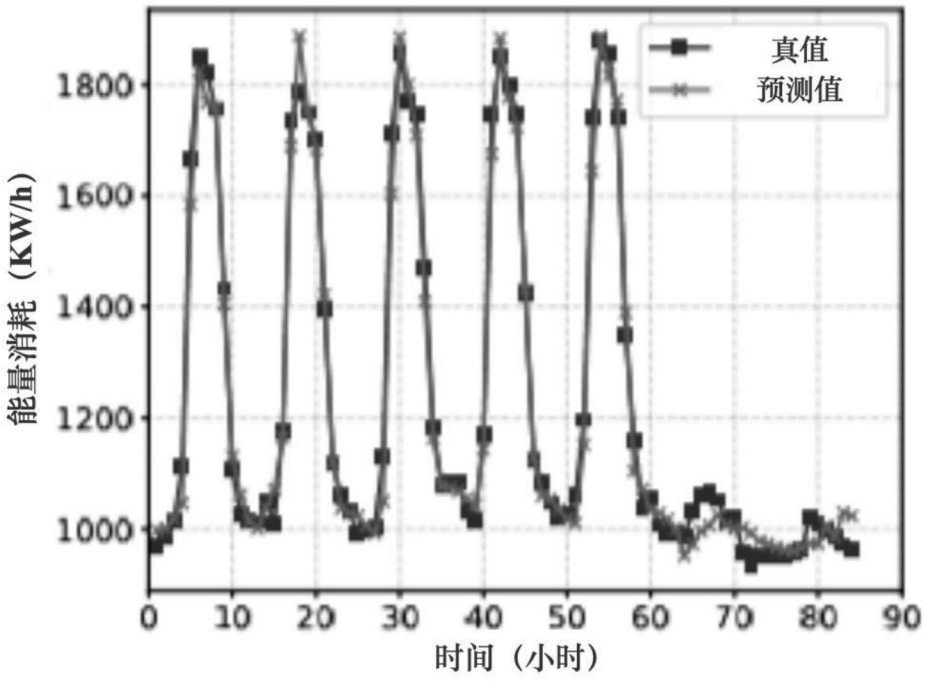

真值
预测值
能量消耗（KW/h）
时间（小时）
1800
1600
1400
1200
1000
0
10
20
30
40
50
60
70
80
90

图6

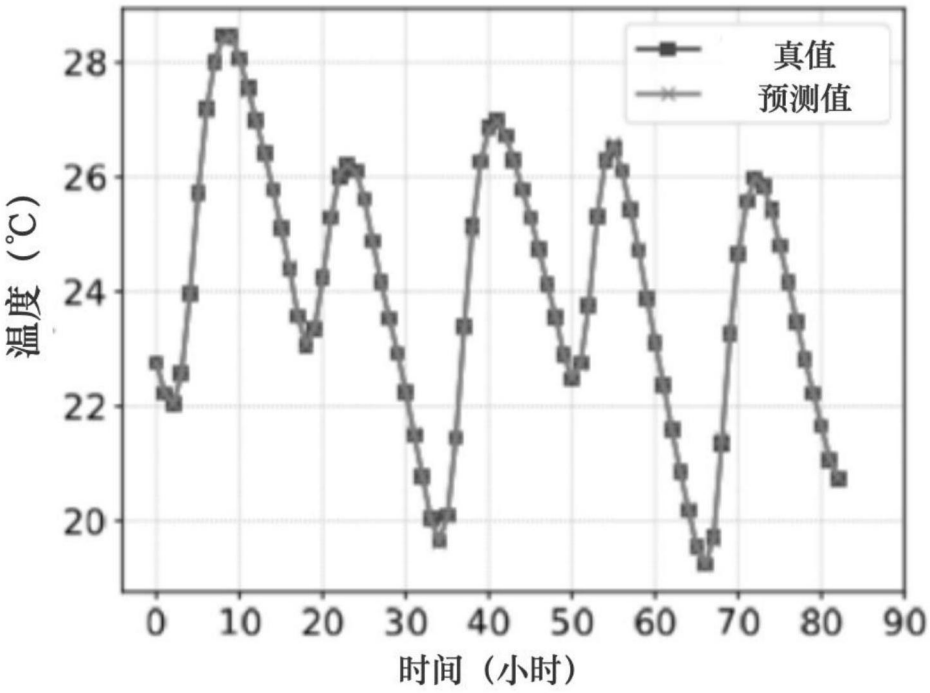

真值
预测值
温度（℃）
时间（小时）
28
26
24
22
20
0
10
20
30
40
50
60
70
80
90

图7

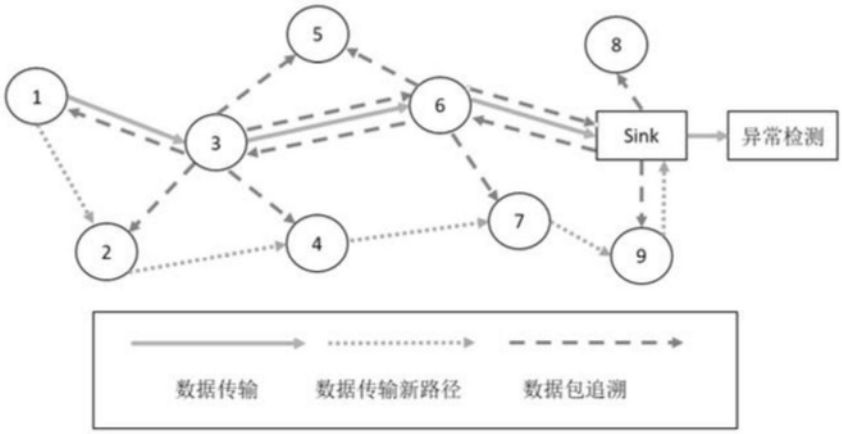

1
2
3
4
5
6
7
8
9
Sink
异常检测
数据传输
数据传输新路径
数据包追溯

图8