

物联网中基于相似性计算的传感器搜索

刘素艳^{*①②} 刘元安^① 吴帆^{①②} 范文浩^{①②}

^①(北京邮电大学电子工程学院 北京 100876)

^②(北京邮电大学安全生产智能监控北京市重点实验室 北京 100876)

摘要: 物联网逐渐成为学术界研究的热点领域,无处不在的传感器设备促进了传感器搜索服务的产生。物联网中搜索的强时空性、海量数据的异构性与传感器节点的资源受限性,给物联网搜索引擎高效地查询传感器提出了挑战。该文提出基于传感器定量数值的线性分段拟合相似性(PLSS)搜索算法。PLSS算法通过分段和线性拟合的方法,构建传感器定量数值的相似性计算模型,从而计算传感器的相似度,根据相似度查找最相似的传感器集群。与模糊集(FUZZY)算法和最小二乘法相比,PLSS算法平均查询精度和查询效率较高。与原数据相比,PLSS算法的存储开销至少降低了两个数量级。

关键词: 物联网; 搜索服务; 传感器搜索; 传感器相似性计算; 线性分段拟合

中图分类号: TP393

文献标识码: A

文章编号: 1009-5896(2018)12-3020-08

DOI: [10.11999/JEIT171085](https://doi.org/10.11999/JEIT171085)

Sensor Search Based on Sensor Similarity Computing in the Internet of Things

LIU Suyan^{①②} LIU Yuanan^① WU Fan^{①②} FAN Wenhao^{①②}

^①(School of Electronic Engineering, Beijing University of Posts and Telecommunications, Beijing 100876, China)

^②(Beijing Key Laboratory of Work Safety Intelligent Monitoring, Beijing University of Posts and Telecommunications, Beijing 100876, China)

Abstract: The Internet of Things (IoT) is becoming a hot research area, and tens of billions of devices are being connected to the Internet which are advancing on the sensor search service. IoT features (searches are strong spatiotemporal variability, limited resources of the sensor, and mass heterogeneous dynamic data) raise a challenge to the search engines for efficiently and effectively searching and selecting the sensors. In this paper, Piecewise-Linear fitting Sensor Similarity (PLSS) search method is proposed. Based on the content values, PLSS calculates the sensor similarity models to search most similarity sensors. PLSS improves the accuracy and efficiency of search compared with FUZZY set algorithm (FUZZY) and least squares method. PLSS storage costs are at least two order of magnitude less than raw data.

Key words: Internet of Things (IoT); Search services; Sensor search; Sensor similarity computing; Piecewise-linear function fitting

收稿日期: 2017-11-20; 改回日期: 2018-09-12; 网络出版: 2018-09-20

*通信作者: 刘素艳 153897455@qq.com

基金项目: 国家自然科学基金(61272518, 61502050), 安全生产智能监控北京市重点实验室主任基金(北京邮电大学), 广东省“扬帆计划”引进创新创业团队项目

Foundation Items: The National Natural Science Foundation of China (61272518, 61502050), The Beijing Key Laboratory Director Foundation of Work Safety Intelligent Monitoring (Beijing University of Posts and Telecommunications), The YangFan Innovative & Entrepreneurial Research Team Project of Guangdong Province

1 引言

迅速发展的网络技术和无处不在的传感器设备,使物联网成为学术界关注的热点领域^[1]。2003年,全球有大约63亿人口,仅有约5亿设备连接到互联网,到2010年,连接到互联网的设备增加到约125亿,而人口仅增加到约68亿^[2]。近年来,无线传感器技术和电信业发展迅速,智能手机作为一种传感器设备,产生了大量的实时数据^[3]。而未来20年,将会有1万亿台物联网感知设备出现,从而生成数十亿的异构数据^[4,5]。

随着大量的传感器接入互联网,海量数据的实时产生,互联网搜索引擎无法满足查找大量非文本传感器数据的需求,这给物联网搜索引擎带来了机遇。搜索的强时空性、海量数据的异构性与传感器节点的资源受限性,使得物联网搜索引擎的研究势在必行^[6,7]。

在传统的互联网中,已有大量的研究人员投入到对海量高维数据相似性搜索的研究中,常见的相似性搜索算法有局部敏感哈希^[8]、K邻近搜索^[9]等。而传统互联网数据大多为相对静态的文本性数据,对于高动态的、异构的与海量的物联网数据,互联网相似性搜索方法不再适用。物联网数据相似性搜索应用广泛,具有很大的研究价值,例如:大量的传感器被部署,来检测森林火灾,观察河流水位,分析道路状况等。但是,部分传感器标识工作需要部署人员来完成,制造商并没有完成,部署和标识大量的传感器,成为一项艰巨的工作。为了解决该问题,传感器自动标识服务查询具有相似输出的传感器,并获取查询到的传感器原数据,自动标识新部署的传感器,从而催生了传感器相似性查找服务。另外,传感器用于感应物理环境和物理实体,通过查找相似的传感器,可以实现对物理环境和物理实体的相似性查找。例如:基于地理位置的地点推荐服务通过相似性搜索可以帮助用户查找其可能感兴趣的地点^[10]。用户需要查找与某个著名景点A具有相同气候的景点去旅行。通过查询传感器的温度与湿度数据值,来分析目标景点是否与A相似,从而辅助用户出行。在智能工业中,传感器相似性查找还可用于机器故障检测,比如:某种故障发生时,设备的温度与压强呈现特定的函数变化规律,根据传感器输出的规律相似性,可以定位设备故障。在智能农业中,传感器相似性查询还可辅助病虫害的定位预防,比如:虫卵孵化需要特定的气候,当气候和土壤温湿度等情况合适时,有很大概率发生某种病虫害。更多其他场景(如:查找适宜

治疗和修养的病房,查找合适的会议室)都依赖于传感器相似性搜索服务。总之,传感器相似性搜索具有很高的研究和应用价值。

目前,国内外针对传感器搜索的研究刚刚起步,主要分为预测和相似性分类查找两个方向。传感器预测方向将传感器输出数据分为定量数据和定性数据,定量数据是感应的原数据(如:温度,湿度),定性数据是传感器的统计或分类信息(如:会议室的占用,空闲状态)^[11]。Elahi等人^[12],基于传感器相关性创建预测模型。Dyser^[13]是一个典型的搜索引擎,提供一个可以匹配搜索请求的物理实体集预测模型。上文提到的研究,都要求传感器输出定性数据,不能应用于输出定量数据的传感器。输出定性数据的传感器,需要有计算或者汇总统计的能力。由于传感器资源或能量的局限性,很多传感器不能输出定性数据。基于传感器定量数据,文献^[14]使用最小二乘多项式拟合来减少传感器的通信开销,但是拟合多项式作为先验条件获取困难。

针对传感器相似性搜索方向的研究,文献^[15]将传感器的相对静态的属性(如:精度,能耗)进行分类,在分类的基础上进行搜索。根据传感器定量数据进行相似性搜索的研究有限,文献^[16]提出FUZZY算法进行传感器相似性搜索,假设传感器能够计算模糊集,从而生成温度概率函数,用于计算传感器的相似性,依据相似性计算数值,获取传感器部署位置信息,对传感器要求较高。基于时间间隔的温度概率函数方法,对查询时间限制严格,比如:查询了时间段 $[t_1, t_2]$ 的数据,并计算了数据密度函数,但是当要查询时间段 $[t_1, t_3]$ 的数据时,即使 $t_3 < t_2$,密度函数仍然需要重新计算。在封闭空间中温度是相似的,所以通过温度概率函数实现传感器相似性搜索的方法,具有较低的准确度。

为了解决上述问题,本文基于传感器动态定量数据,采用PLSS算法(Piecewise-Linear fitting Sensor Similarity search method)计算传感器间的相似值,并按照相似值进行查找结果匹配排序,完成物联网中传感器相似性搜索。首先,本文根据传感器数据集找到数据函数变化的分段点,将数据分成不同的子集;然后对数据子集进行线性分段拟合,得到传感器数据的时间连续性拟合模型;再定义传感器相似度计算方法;最后根据匹配排序算法,将相似性计算结果进行排序,完成传感器相似性查询。

2 物联网传感器搜索框架

本节描述传感器搜索的体系结构。在资源受限

的物联网中,传感器节点的能力有限,遍历搜索模式将给传感器节点带来严重的通信和能源成本。因此,基于传感器原始数据,设计一个低成本的搜索框架,提高搜索效率非常重要。

如图1所示,传感器定期收集物理实体的状态信息,自动标识并上报底层网关。物联网中海量的传感器数据存储格式各异,使得数据处理和利用困难,异构数据格式统一越来越受到重视^[17]。本文定义传感器定量数据的时间序列模型为 (X_{t_i}, Y_{t_i}) , X_{t_i} 代表时间点, Y_{t_i} 代表传感器感应物理环境的值, t_i 代表时间序列。随着时间的推移,传感器每个动态属性都生成了大量时间相关的数据。根据传感器数据,底层网关为每个传感器创建相似性计算模型,并将其存储在相似模型数据库中。传感器的相似模型在设计和构建上相互独立,可通过并行计算层来计算相似值。当用户通过客户端提交一个查询需求时,并行计算层通过用户需求获取目标查询模型,并根据目标查询模型计算相似的传感器模型序列,最后根据结果序列排序,反馈检索结果。本文给出查询模型 $q = (t_q, a_i, F(X), (R_q))$, 其中 t_q 表示传感器搜索的时间区间, a_i 代表特征属性, $F(X)$ 代表相似性计算模型, R_q 代表查询结果。PLSS算法的搜索流程如图2所示。

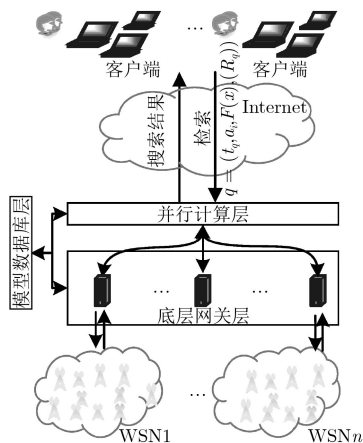


图1 物联网基于内容的传感器查询体系架构

3 PLSS算法

要实现传感器相似性计算,首先假设传感器按时间间隔采集物理实体状态并自动上报本地网关,根据数据拟合函数计算传感器的相似性。因此,计算传感器的相似性,关键是找到数据拟合的方法。本文提出PLSS算法,基于传感器定量数据,解决了使用定性数据对传感器资源要求较高的问题;采用分段线性拟合方法,避免了最小二乘拟合多项式获得困难的问题;时间连续的相似性计算模型,提高了查找的灵活性,从而解决了FUZZY算法对查

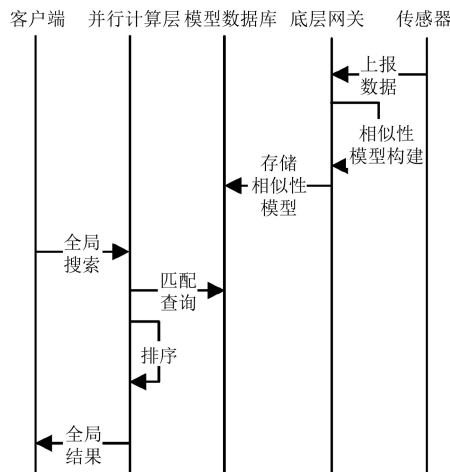


图2 PLSS搜索流程

找时间区间的强依赖性问题。

3.1 数据子集中分段点的计算方法

要实现传感器数据的数据拟合,首先要对数据进行分段,将传感器数据集分成子数据集,针对子数据集进行线性拟合。如图3曲线依据 Δt 分割数据区间,当 $\Delta t \rightarrow 0$ 时,总能使得时间区间 $[t_i, t_{i+1}]$ 内的曲线函数近似成直线函数。因此,本文假设传感器动态属性函数在相对小的时间区间内,传感器属性值 Y 与时间 X 为直线函数:

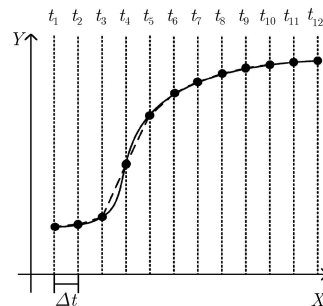


图3 数据分段示意图

$$Y = a + bX, a, b \in N \quad (1)$$

本节主要介绍数据分段的方法,数据分段关键是找到传感器属性值的分段点,使得在分段数据子集内,传感器函数可近似拟合为直线函数。本文提出如下分段点的计算方法:

第1步 设置阈值 δ , 其中 δ 定义为

$$\delta = \varepsilon \times (Y_{t_1} + Y_{t_2} + \dots + Y_{t_n}) / n, 0 < \varepsilon < 100\% \quad (2)$$

其中, n 为传感器值采样个数, t_i 为采样时间序列, ε 表示拟合数值与算数平均值的偏移率, ε 越大直线分段点越少, 存储开销越小, 查询精度越低, 反之 ε 越小存储开销越大, 查询精度越高;

第2步 底层网关根据传感器初始数值, 取 j 个数据 $[t_1, t_j]$, 生成直线函数, 并将分段点计入数据集为

$$I = \{(X_{t_i}, Y_{t_i})\} \quad (3)$$

第3步 将传感器数值 $(X_{t_{j+1}}, Y_{t_{j+1}})$ 代入第2步的直线函数, 计算差值

$$\Delta Y = |Y_{t_{j+1}} - a - bX_{t_{j+1}}| \quad (4)$$

如果 ΔY 大于阈域值 δ , 将 $(X_{t_{j+1}}, Y_{t_{j+1}})$ 计入数据集 I , 得到

$$I = \{(X_{t_i}, Y_{t_i}), (X_{t_{j+1}}, Y_{t_{j+1}})\} \quad (5)$$

剩余的数据值重复步骤2。否则, 计算下一个数值 $(X_{t_{j+2}}, Y_{t_{j+2}})$, 重复第3步;

第4步 根据数据集 I 内的分段点, 生成数据子集。假设生成了 m 段的时间子集, 如式(6):

$$T = \{T_i | 1 \leq i \leq m\}, T_i = [t_i, t_j] \quad (6)$$

其中, 数据分段点子集 $\{(X_{t_i}, Y_{t_i}), (X_{t_j}, Y_{t_j})\} \in I$, t_i 代表第 i 个时间序列, T_i 代表第 i 个时间段。

3.2 数据子集的线性拟合算法

本文主要介绍传感器数据子集的线性拟合方法。根据3.1节分段点获得传感器数据子集 T , 作为数据矩阵式(7)的数据行, 进行线性拟合。

采集数据的方法为, 在每个时间段内, 采集 k 个数据点, 从而获得了 m 行数据, (X_{t_i}, Y_{t_i}) 标识数据点, 用数据矩阵表示:

$$M = \begin{bmatrix} X_{t_{11}}, Y_{t_{11}} & X_{t_{12}}, Y_{t_{12}} & \cdots & X_{t_{1k}}, Y_{t_{1k}} \\ X_{t_{21}}, Y_{t_{21}} & X_{t_{22}}, Y_{t_{22}} & \cdots & X_{t_{2k}}, Y_{t_{2k}} \\ \vdots & \vdots & \ddots & \vdots \\ X_{t_{m1}}, Y_{t_{m1}} & X_{t_{m2}}, Y_{t_{m2}} & \cdots & X_{t_{mk}}, Y_{t_{mk}} \end{bmatrix} \quad (7)$$

用 Δt 作为时间间隔, 将 m 维动态属性时间数据集 $T = \{T_i | 1 \leq i \leq m\}$ 分成离散的时间点 $\{t_i | i = 1, 2, \dots, k\}$, t_i 代表时间序列。时间序列满足式(8):

$$t_{i+1} = t_i + \Delta t, i \in \{1, 2, \dots, k-1\}, k \in N^* \quad (8)$$

对于每一行的数据, 用线性拟合算法得到 m 个等式如式(9):

$$\left. \begin{aligned} g_1(x) &= a_1 + b_1x, X_{t_{11}} \leq x \leq X_{t_{1k}} \\ g_2(x) &= a_2 + b_2x, X_{t_{21}} \leq x \leq X_{t_{2k}} \\ &\vdots \\ g_m(x) &= a_m + b_mx, X_{t_{m1}} \leq x \leq X_{t_{mk}} \end{aligned} \right\} \quad (9)$$

因此, 在整个数据区间内获得了 m 个分段直线函数。本文使用函数左连续处理式(9)得到

$$F(x) = \begin{cases} g_1(x) = a_1 + b_1x, X_{t_{11}} \leq x < X_{t_{1k}} \\ g_2(x) = a_2 + b_2x, X_{t_{1k}} \leq x < X_{t_{2k}} \\ \vdots \\ g_m(x) = a_m + b_mx, X_{t_{m-1k}} \leq x \leq X_{t_{mk}} \end{cases} \quad (10)$$

式(10)是数据的线性拟合函数。

3.3 相似性模型的构建方法

本节详细描述传感器相似性计算模型的拟合过程。

首先, 对数据进行去量纲处理, 其中 y_{t_i} 表示原始属性值, $\bar{y}$ 表示属性值的算术平均值, 如式(11):

$$y = y_{t_i} / \bar{y} \quad (11)$$

数据处理后传感器 b 的特征 a 数据集为

$$P_b^a = \{(x_{t_i}, y_{t_i}) | x_{t_i} \in T, y_{t_i} \in R, 1 \leq t_i \leq n, i \in N^*\} \quad (12)$$

其中, 定义时间区间 T , x_{t_i} 为时间数据, y_{t_i} 为传感器属性值, t_i 为时间序列。定义时间等式关系为

$$x_j = x_i + (t_j - t_i) \cdot \Delta t \quad (13)$$

然后, 使用3.1节方法, 获得式(14)所示的数据子集:

$$p_b^a = \{p_m^a | 1 \leq m \leq n, p_m^a \neq \emptyset\}, \sum_{i=1}^m |p_i^a| = n \quad (14)$$

其中, m 表示数据分组数量, $|p_i^a|$ 表示子集内数据个数, 数据总量为 n 。然后, 根据式(9), 首先将

$$D = \{p_i^a | i \in 1, 2, \dots, m\} \quad (15)$$

拟合为数据函数集:

$$F = \{f_i^a(X) | X \in T, i \in 1, 2, \dots, m\} \quad (16)$$

然后, 根据式(10)传感器属性计算模型拟合如式(17):

$$F_b^a(X) = \sum_{i=1}^m f_i^a(X), i = 1, 2, \dots, m, X \in T \quad (17)$$

4 传感器查询匹配算法

本节给出依据相似性模型计算传感器相似度的计算方法, 再根据相似度进行查询匹配排序, 从而实现传感器的相似性查询。

4.1 相似度的计算方法

给定传感器 S_t 为查询目标传感器, 计算传感器 $\{S_i | i \in N^*\}$ 分别与目标传感器的相似性。依据3.3节传感器相似性模型的构建方法, 分别获得特征 h 的模型函数 $\{F_i^h(X) | i \in N^*\}$ 。

依据 S_t 定期采集传感器特征 h 的时间序列属性值 $\{(X_{t_1}, Y_{t_1}), \dots, (X_{t_n}, Y_{t_n})\}$, 根据式(18)计算模型与目标传感器之间的绝对差:

$$Ad_i = \sum_{i=1}^m \sum_{j=1}^n |F_i^h - Y_{t_j}| \quad (18)$$

m 标识传感器的个数, n 标识传感器属性值时间序列的个数, Ad_i 代表第 i 个传感器与目标传感器的相似性。将绝对差进行降序排列, 绝对差越小模型之间的相似度越高。

4.2 匹配排序算法

在4.1节中提出模型相似值 Ad 的计算方法, 本节将给出根据相似度计算查询结果的方法。并行计

算模型相似性, 计算结果具有大量性和无序性, 为了提高查询准确度和效率, 本文结合桶排序和交换排序的原理, 对查询结果进行排序。桶排序首先对结果进行分组, 在组内利用交换排序排除不符合要求的结果, 从而对结果集进行快速筛选。下文给出查询结果的匹配排序计算方法:

首先, 对Ad进行互斥性分组, 得到

$$G = \{g_1, g_2, \dots, g_q\}, q \in N^*, \sum_{j=1}^q |g_j| = m \quad (19)$$

其中, g_j 表示第 j 个分组, $|g_j|$ 代表 j 分组内的相似值Ad的数据个数, m 标识参与相似性计算的传感器模型个数。

然后, 定义查询基准区间 $|\varepsilon|$, 相似度在区间内, 标识查询准确, 不在区间内标识查询不准确, 分别对 g_j 依据交换排序原理进行排序, 如式(20):

$$R_q = \sum_{j=1}^q \begin{cases} R_i^j, \text{Ad} \in g_j, \text{Ad} \leq |\varepsilon|, 1 \leq j \leq m \\ \emptyset, \text{其它} \end{cases} \quad (20)$$

其中, R_i^j 表示通过 g_j 组Ad计算的结果集。

最后, 依次输出传感器查询结果列表 R_q 。

5 仿真验证

本文使用Intel Berkeley [18]数据集进行评估测试, 计算机仿真环境为Matlab R2015b。在仿真时, 相同传感器属性统一单位。传感器1的数据集中不包含噪音点, 而传感器20的数据中明显的包括噪音点。而且, 传感器1和传感器20的温度数值区分度不大, 有利于测试PLSS算法与FUZZY算法的差异。因此, 本文采用传感器1和传感器20的温度数值进行验证测试。传感器1的数据集取1317个输出值。传感器20数据集取2059个输出值。围绕传感器1和传感器20分别取4个传感器节点作为查询验证点, 如图4所示。

5.1 查询准确度对比

空气是个连通体, 地理位置相近的传感器采集的温度值应该相近。因此, 本文将传感器1和传感器20作为查询的基准传感器, 传感器2, 17, 19, 21, 22, 32, 33, 35分别取10组数据计算与传感器1和传

感器20的相似度。如图4所示, 传感器2, 32, 33, 35靠近传感器1, 如果计算结果与传感器1更相似, 则赋值查询结果为1标识查询准确; 反之为0标识查询错误; 与传感器1和传感器20的相似度相等, 则赋值为0.5标识查询结果不确定。传感器17, 19, 21, 22靠近传感器20, 查询结果规则同上, 计算结果与传感器20更相似查询结果赋值1, 否则0, 相等0.5。最后, 将每个传感器节点的10组查询结果求平均值, 得图5中的查询准确度。

FUZZY算法没有介绍数据噪音点的处理流程, 属性取值区间的计算是属性最大值与最小值的差, 会受到噪音点的影响, 从而影响相似系数的计算。在文献[16]中提到, 当属性值相似时, FUZZY算法查询效果相对较差, 查询准确度不高。PLSS算法的查询准确度在 ε 取1%时查询准确率高于FUZZY算法, 随着 ε 取值的增大PLSS算法的查询准确率有所下降, 对于查询准确度都不高(小于0.6)的传感器节点, 即使PLSS算法的 ε 取5%时, 查询准确率仍然高于FUZZY算法, 对于查询准确率相对较高(大于0.8)的传感器节点, PLSS算法的查询准确度略低于FUZZY算法, 但是当 ε 取5%时, 总平均查询准确率仍然高于FUZZY算法, 如图5(a)所示。

最小二乘直线拟合, 数据拟合不足, 查询准确度稳定性差, 传感器2, 33, 35的准确率小于等于0.1, 平均查询准确度明显低于PLSS算法, 如图5(b)所示。文献[14]利用最小二乘多项式做传感器数据拟合, 拟合多项式获取困难。为了解决上述问题, 并且避免拟合不足和过拟合现象, 本文取查询平均准确度大于等于60%的最低次幂多项式作为最小二乘法的拟合多项式进行相似性查询, 查询准确度如图5(c)所示, 当 ε 取5%与1%时PLSS查询准确度均大于等于最小二乘拟合算法。

5.2 查询效率对比

查询效率是搜索引擎另一个重要的指标, 本节主要分析PLSS算法的查询效率。假设计算资源和计算环境对查询时间无影响, 在相同的运行环境中取相同数据进行比对, 并且比对时间取查询10次的平均时间用于减小误差。

FUZZY算法查询对时间要求严格, 使得每次查询必须重新计算密度函数, 因此查询不能并行进行, 本文进行8个传感器的80组查询, 则FUZZY算法需要计算80组密度函数, 而PLSS算法以时间序列作为参数, 有效地解决了时间局限性的问题, 对于80组传感器的查询, 只需要对目标传感器1和传感器20做1组分段拟合函数, 因此PLSS算法的查询可以并行进行, PLSS算法的查询效率明显高于

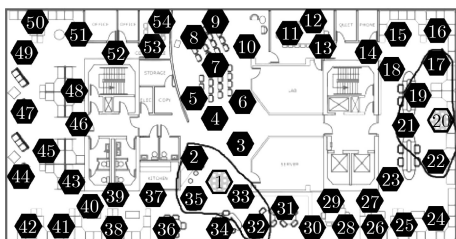


图 4 Intel Berkeley 传感器分布图

FUZZY算法，如图6(a)所示。

传感器的数值可以表示为时间序列的最小二乘拟合多项式^[14]，查询时间与数据量 n 和多项式最高次幂 m 有关。PLSS算法得到的是直线函数，查询时间只取决于查询数据量 n 。本文80组数据查询的效率，PLSS算法高于最小二乘拟合算法，如图6(b)所示。

5.3 存储开销对比分析

大量的传感器连接到物联网，海量数据的存储与传递给有限的网络资源带来了挑战。通过减少传感器数据的存储开销来减少海量数据对网络的压力，成为当下研究的热点问题^[19]。

传感器原数据的存储消耗了大量的存储空间，

大量数据的传递也给有限的网络资源形成了巨大的压力。将原数据拟合为曲线函数，存储和传递曲线函数，只需要存储函数的系数和变量的取值区间，比如： $y = ax^2 + b, x \in [1, 5]$ ，只需存储次数2，系数 a, b ，变量区间1, 5。FUZZY算法、最小二乘法与PLSS算法都可以通过此方法节省存储开销。FUZZY与PLSS两种方法存储的都是直线，因此只需要存储系数和变量区间共4个维度的数据。最小二乘存储的是多项式，存储系数与多项式的幂有关，本文拟合多项式最高次幂为6，因此存储系数和变量区间共9个维度的数据。

表1为数据的存储对比。本文假设存储数据都是浮点数，占用空间一致。其中传感器数据包括时

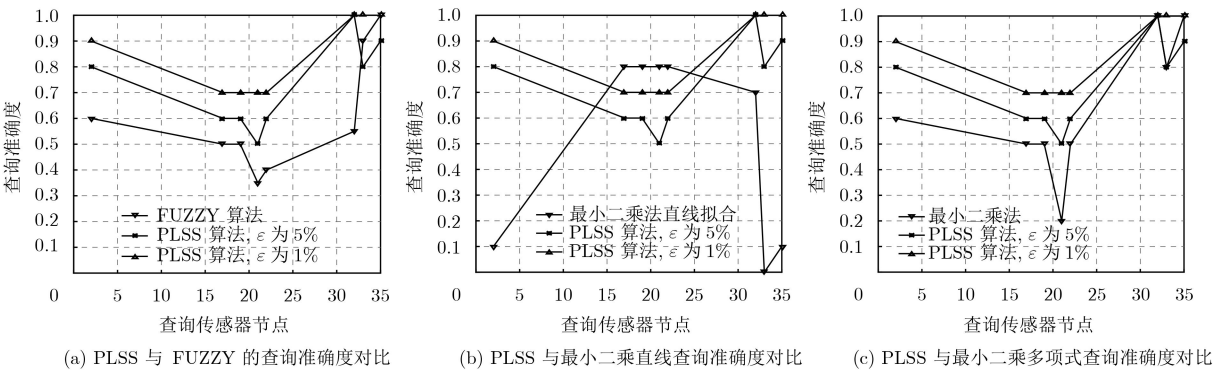


图5 查询准确度对比

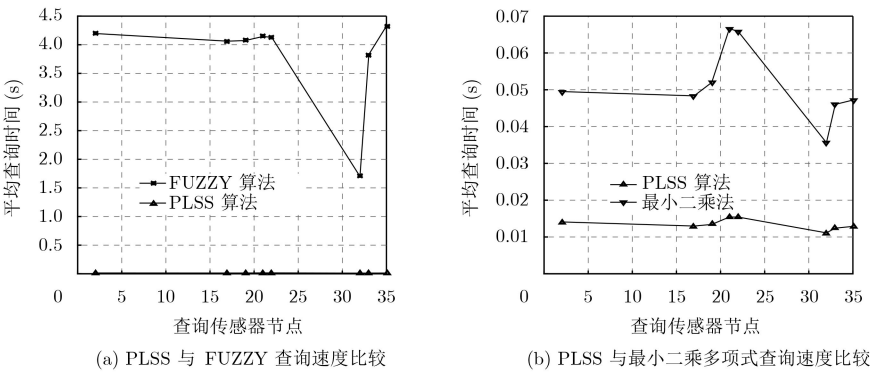


图6 查询速度比较

表1 数据存储开销分析

	传感器1	传感器20	数据个数统计
原数据 (时间, 传感器值)	1317×2	2059×2	6.752×10 ³
FUZZY算法 (传感器平均数据密度函数)	16×4×10	20×4×10	2.400×10 ³
FUZZY算法 (传感器平均数据斜率密度函数)	10×4×10	14×4×10	
最小二乘多项式拟合算法 (传感器函数系数)	9	9	1.800×10
PLSS算法 (传感器函数系数)	16	25	4.100×10

间和传感器值2个维度,因此原数据的存储占用空间为数据个数的2倍。本文每个传感器节点分别取10组数据进行对比,FUZZY算法查询时间段的局限性使得10组数据需要分别做包括数据密度和数据斜率密度的拟合,因此存储空间为10组数据密度拟合函数系数个数加上斜率密度函数系数个数的和,再乘上10。比如:对于1个传感器节点,每组数据中平均数据密度函数为16个,平均斜率密度函数为10个,那么对于1个传感器中1组数据的存储空间为 $(16+10) \times 4$,10组的存储空间为 $(16+10) \times 4 \times 10$ 。通过计算,FUZZY算法与原数据对存储的开销数量级都是 10^3 。最小二乘多项式拟合与PLSS算法以时间为参数,只需要拟合一次函数,可进行不同时间段的多次查询。对于PLSS算法 ε 取值越小拟合直线越多,当 ε 取1%时,PLSS算法存储空间与原数据和FUZZY算法相比仍然降低了2个数量级。

6 结束语

本文基于传感器时间序列定量属性值计算传感器的相似性,避免了传感器资源受限性问题。相比使用FUZZY算法与最小二乘法分析传感器的相似性,PLSS算法的查询准确度较高。PLSS算法以时间为拟合参数的分段拟合方法,解决了FUZZY算法中查找时间区间强依赖性的问题,使得依据时间子集查询数据的灵活性增加,解决了最小二乘法拟合多项式获取困难的问题,查询效率也得到大幅度提高。PLSS算法相对于FUZZY算法与传感器原数据,存储开销降低了两个数量级。查询计算耗时与数据模型和资源开销等都有很大关系,未来项目组将针对资源开销和数据模型复杂度与查询计算的关系,做进一步的研究。

参考文献

- [1] BELLO O and ZEADALLY S. Intelligent device-to-device communication in the internet of things[J]. *IEEE Systems Journal*, 2016, 10(3): 1172–1182. doi: [10.1109/JSYST.2014.2298837](#).
- [2] EVANS D. The Internet of things: How the next evolution of the internet is changing everything[C]. Cisco Internet Business Solutions Group, San Francisco, USA, 2011.
- [3] RIBEIRO M, GROLINGER K, and CAPRETZ M A M. MLaaS: Machine learning as a service[C]. 2015 IEEE 14th International Conference on Machine Learning and Applications., Miami, USA, 2015: 896–902.
- [4] LI Shancang, LI Daxu, and ZHAO Shanshan. The internet of things: A survey[J]. *Information Systems Frontiers*, 2015, 17(2): 243–259. doi: [10.1007/s10796-014-9492-7](#).
- [5] 张普宁. 面向物联网搜索服务的实体状态匹配估计方法研究[D]. [博士学位], 北京邮电大学, 2017.
- [6] ZHANG Puning. Research on entity state matching estimation method towards search service in the internet of things[D]. [Ph.D. dissertation], Beijing University of Posts and Telecommunications, 2017.
- [7] 于海宁, 张宏莉, 方滨兴, 等. 物联网中物理实体搜索服务的研究[J]. *电信科学*, 2012, 28(10): 111–119. doi: [10.3969/j.issn.1000-0801.2012.10.019](#).
- [8] YU Haining, ZHANG Hongli, FANG Binxing, et al. Research on search service for physical entities in the internet of things[J]. *Telecommunications Science*, 2012, 28(10): 111–119. doi: [10.3969/j.issn.1000-0801.2012.10.019](#).
- [9] WU Dapeng, HE Jing, WANG Hongguang, et al. A hierarchical packet forwarding mechanism for energy harvesting wireless sensor networks[J]. *IEEE Communication Magazine*, 2015, 53(8): 92–98. doi: [10.1109/MCOM.2015.7180514](#).
- [10] LI Dongsheng, ZHANG Wanxin, SHEN Siqi, et al. SLSH: Shuffle-efficient locality sensitive hashing for distributed similarity search[C]. 2017 IEEE 24th International Conference on Web Services, Honolulu, USA, 2017: 822–827.
- [11] ZHAO Xujun, ZHANG Jifu, and QIN Xiao. kNN-DP: Handling data skewness in kNN joins using mapreduce[J]. *IEEE Transactions on Parallel and Distributed Systems*, 2017, 29(3): 600–613. doi: [10.1109/TPDS.2017.2767596](#).
- [12] 蒋翠清, 疏得友, 段锐. 基于用户时空相似性的位置推荐算法[J]. *计算机工程*, 2018, 44(7): 177–182. doi: [10.19678/j.issn.1000-3428.0047996](#).
- [13] JIANG Cuiqing, SHU Deyou, and DUAN Rui. Location recommendation algorithm based on spatial-temporal similarity of user[J]. *Computer Engineering*, 2018, 44(7): 177–182. doi: [10.19678/j.issn.1000-3428.0047996](#).
- [14] 张普宁, 刘元安, 吴帆, 等. 物联网中适用于内容搜索的实体状态匹配预测方法[J]. *电子与信息学报*, 2015, 37(12): 2815–2820. doi: [10.11999/JEIT150191](#).
- [15] ZHANG Puning, LIU Yuanan, WU Fan, et al. An entity state matching prediction method for content-based search in the internet of things[J]. *Journal of Electronics & Information Technology*, 2015, 37(12): 2815–2820. doi: [10.11999/JEIT150191](#).
- [16] ELAHI B M, ROMER K, OSTERMAIER B, et al. Sensor ranking: A primitive for efficient content-based sensor search[C]. International Conference on Information Processing in Sensor Networks, San Francisco, USA, 2009: 217–228.
- [17] ROMER K, OSTERMAIER B, OSTERMAIER F, et al. Real-time search for real-world entities: A survey[J]. *Proceedings of the IEEE*, 2010, 98(11): 1887–1902. doi: [10.1109/JPROC.2010.2062470](#).

- [14] ZHANG Puning, LIU Yanan, WU Fan, *et al.* Low-overhead and high-precision prediction model for content-based sensor search in the internet of things[J]. *IEEE Communications Letters*, 2016, 20(4): 720–723. doi: [10.1109/LCOMM.2016.2521735](https://doi.org/10.1109/LCOMM.2016.2521735).
- [15] EBRAHIMI M, SHAFIEIBAVANI E, WONG R K, *et al.* An adaptive meta-heuristic search for the internet of things[J]. *Future Generation Computer Systems*, 2017, 76(11): 486–494.
- [16] TRUONG C, ROMER K, and CHEN K. Fuzzy-based sensor search in the web of things[C]. 2012 3rd International Conference on the Internet of Things, Wuxi, China, 2012: 127–134.
- [17] ZHUKOV V and KOMAROV M. Semantic control method of the internet of things based on linked open data[C]. 2017 IEEE 19th Conference on Business Informatics, Thessaloniki, Greece, 2017: 1–4.
- [18] Intel Berkeley Research lab. Intel berkeley research lab sensors data[OL]. <http://db.csail.mit.edu/labdata/labdata.html>. 2004.10.
- [19] DIAS G M, BELLALTA B, and OECHSNER S. A survey about prediction-based data reduction in wireless sensor networks[J]. *ACM Computing Surveys*, 2016, 49(3): 58.
- 刘素艳：女，1982年生，博士生，研究方向为物联网搜索、无线传感器网络。
- 刘元安：男，1963年生，教授，研究方向为电磁兼容、泛在无线网络。
- 吴帆：女，1981年生，副教授，研究方向为物联网搜索、泛在无线网络。
- 范文浩：男，1986年生，讲师，研究方向为移动设备、云计算。