

DeepHome: 一种基于深度学习的智能家居管控模型

毛 博¹⁾ 徐 恪^{2),3)} 金跃辉¹⁾ 王晓亮^{2),3)}

¹⁾(北京邮电大学网络与交换技术国家重点实验室 北京 100876)

²⁾(清华大学计算机科学与技术系 北京 100083)

³⁾(清华大学信息科学与技术国家实验室 北京 100083)

摘 要 作为物联网技术在日常生活领域的重要应用,智能家居产业近年来取得了快速发展.但是不同智能家居设备之间差异化的通信方式与割裂的功能,大大增加了用户管理与使用的复杂度.智能家居管控平台旨在整合异构网络环境下多种设备的数据监测与控制能力,为用户提供整体化家居服务.在面对多种设备协同、环境参数众多、用户需求难以确定等众多困难时,如何建立一种无感化、精确化、智能化的家居设备统一管控能力,是智能家居平台化过程中亟需解决的问题.为破解智能家居设备自动管控难题,文中基于深度学习方法提出 DeepHome 智能家居管控模型. DeepHome 模型首先采用自编码网络构建设备模型,通过逐层无监督预训练,挖掘通用化设备特征;继而基于具体家居场景,综合多个独立设备模型构建多隐层学习网络,并使用家居环境数据进行模型整体训练.经过训练,DeepHome 模型能够基于家居环境数据预测智能设备工作状态,并依照预测结果调整相应设备,实现对智能家居设备的自动化统一管控.由于智能家居平台尚处于起步阶段,现阶段仍缺乏能够有效描述家居环境整体的数据.文中设计调查问卷与数据收集网站收集不同用户的家居环境、设备数据与生活习惯,并基于所得数据构建智能家居环境仿真测试平台 HomeTest,模拟生成批量家居环境数据,辅助模型训练.使用仿真数据进行 10 轮训练后,DeepHome 模型对智能设备工作状态的预测准确率达到 99.4%,较浅层神经网络模型提高了 6.4%,较基于逻辑规则的控制方案提高了 36.1%;与此同时,DeepHome 在设备状态需要调整时的预测准确率达到 74.1%,较浅层神经网络模型提高了 4.7 倍,较基于逻辑规则的控制的方案提高了 13.2 倍.在真实环境数据集下,DeepHome 的对设备状态的预测准确率也达到了可以被用户接受的 98.9%.实验表明,DeepHome 模型能够充分发掘智能设备状态与环境、用户以及其他设备间的联动关系,实现对设备的智能化控制.

关键词 智能家居;深度学习;机器学习;自编码网络;物联网

中图法分类号 TP18 **DOI 号** 10.11897/SP.J.1016.2018.02689

DeepHome: A Control Model of Smart Home Based on Deep Learning

MAO Bo¹⁾ XU Ke^{2),3)} JIN Yue-Hui¹⁾ WANG Xiao-Liang^{2),3)}

¹⁾(State Key Laboratory of Networking and Switching Technology, Beijing University of Posts and Telecommunications, Beijing 100876)

²⁾(Department of Computer Science and Technology, Tsinghua University, Beijing 100083)

³⁾(National Laboratory for Information Science and Technology, Tsinghua University, Beijing 100083)

Abstract As an important application of the Internet of things technology in the field of daily life, the smart home industry has made rapid development in recent years. However, the independent communications and the fragmented functions between different smart home devices increase the complexity of user management and use. Smart home control platform is designed to integrate monitoring data and control capabilities from lots of smart home devices and to provide

收稿日期:2016-07-10;在线出版日期:2017-06-15. 本课题得到国家自然科学基金(61170292, 61472212)、国家科技重大专项课题(2015ZX03003004)、国家“九七三”重点基础研究发展规划项目基金(2012CB315803)、国家“八六三”高技术研究发展计划项目基金(2013AA013302, 2015AA015601)、欧盟 CROWN 基金项目(FP7-PEOPLE-2013-IRSES-610524)、清华信息科学与技术国家实验室(筹)学科交叉基金项目资助. 毛 博,男,1991 年生,硕士,主要研究方向为无线网络技术、无线传感器网络应用、机器学习. E-mail: maobo1991@163.com. 徐 恪,男,1974 年生,博士,教授,博士生导师,中国计算机学会(CCF)会员,主要研究领域为计算机网络体系结构、高性能路由器、P2P 网络、物联网、网络经济学. 金跃辉,女,1965 年生,教授,中国计算机学会(CCF)会员,主要研究领域为互联网性能测量与用户体验、社会化网络与推荐系统. 王晓亮,男,1986 年生,博士研究生,主要研究方向为无线网络与移动计算.

integrated home services in heterogeneous network environment. To establish an automated, accurate and intelligent control model of smart home, we have to face the challenges of the coordination of multiple devices, the large number of environment parameters and the uncertainty of user requirements, which is a core problem in the process of smart home platformization. In order to solve the problem of automatic control of smart home devices, we propose DeepHome, a smart home control model based on deep learning algorithm in this paper. At first, the DeepHome model built device model based on autoencoder network model. Secondly, built the universal features of device model by unsupervised training layer by layer. Thirdly, integrated multi independent device model to build a multi hidden layer neural network and trained the complete model by stochastic gradient descent algorithm based on the dataset of home environment. After training, DeepHome model can predict the proper working status of smart devices and adjust the status by predictions, so as to control the smart home devices automatically and uniformly. Due to the smart home platform is still in its infancy, there is not enough effective data to describe the complete home scenarios currently. In this paper, we collected environment data, smart devices data and lifestyles from different users by questionnaires and a website we had established. Based on these data, we designed HomeTest, a simulation platform of smart home environment to generate more data of smart home to train and test the DeepHome model. After 10 rounds of training based on the data from the HomeTest simulation environment, the prediction accuracy rate of smart home devices' working status reaches 99.4% in DeepHome model, which is 6.4% higher than the accuracy rate of shallow learning model, and 36.1% higher than the accuracy rate of control method based on manual logical rules. When the smart home devices need to be changed, the prediction accuracy rate reaches 74.1% in DeepHome model, which is 4.7 times higher than shallow learning model and 13.2 times higher than control method based on manual logical rules. In the real home scenarios, the prediction accuracy rate of the device status reaches 98.9% in DeepHome model, which can be accepted by most users. Experiments show that DeepHome model can exploit the relationships between the device status and the data from home environment, users and other devices, and control smart home devices with intelligence.

Keywords smart home; deep learning; machine learning; autoencoder; Internet of Things

1 引 言

近年来,随着移动互联网的快速发展以及智能硬件、可穿戴设备等物联网设备在人们日常生活中的逐渐普及,“智能化”的概念受到更多消费者的青睐。作为物联网技术在家庭和生活环境中的一个重要应用,智能家居受到了产业界、学术界的广泛关注。智能家居主要以住宅为平台,通过物联网技术连接家居环境中的设备,提供家电控制、远程控制、环境监测、危险预警、安全监控等多种智能化家庭服务。

2014 年底,腾讯家电发起一项针对智能家居的线上、线下调查,调查结果显示,95.19%的受访者表示对智能家居感兴趣,但与此形成鲜明对比的是,87.53%的受访者表示对智能家居的现状不满,认为

其并未达到预期效果^①。学术界近年的研究也在关注智能家居发展现状与其远景设想之间较大落差的成因。一个重要原因在于智能家居厂商与产品的零散与不成体系,独立的硬件设备和割裂的单个功能使用户难以体会智能家居大概概念的全貌^[1-3]。

为整合智能家居资源,提供整体化的智能家居服务,多任务、智能化与平台化逐渐成为智能家居行业的发展方向^[4]。家居设备的智能化自动管控是智能家居平台的核心能力。理想的智能家居自动管控系统需要具备 3 个基本特征:(1)能够在不打扰用户的前提下进行家居环境的自动管控,将用户从繁琐的设备操作中解放出来,即“服务无感化”;(2)能

^① 李坤. 2014 年国内智能家居市场发展报告 [EB/OL]. <http://digi.tech.qq.com/a/20141231/005577.htm>

够在复杂家居环境中准确预测并调整智能设备工作状态,避免设备误操作,即“服务精确化”; (3) 能够充分发掘用户习惯,理解用户需求,优化用户体验,即“服务智能化”。

当前智能家居平台的自动管控技术依然停留在基于用户预设规则进行控制的阶段,繁琐的规则输入增加了设备的使用复杂度与学习成本,僵化的逻辑控制更难以适应多变的家居环境。因此,智能家居平台亟需一种无感化、精确化、智能化的家居设备统一管控模型。

在人工智能领域中,机器学习特别是深度学习方法近年来取得了引人注目的成果^[5-7]。与传统技术相比,深度学习方法大幅提升了语音识别、图像识别、自然语言处理的性能,拥有良好的发展和应用前景^[8]。本文基于深度学习方法提出了一种家居智能化感知控制模型 DeepHome。该模型基于智能家居环境数据进行深度神经网络模型训练,对智能家居用户行为习惯进行感知分析,实现智能家居环境的自动化集中控制。

本文的主要贡献如下:

(1) 通过实验,发现基于逻辑规则的控制方法、基于浅层神经网络模型的控制方法在智能家居环境中的不足。

(2) 面对缺乏家庭环境设备使用数据的问题,提出了一种可行的数据收集与扩充方案。

(3) 首次面向智能家居集中管控的需求和其自身特点,提出了基于深度学习方法的 DeepHome 模型,并通过实验验证了其准确性和有效性。

本文第2节介绍现阶段家居智能化自动控制的相关研究工作;第3节论述本研究相关的智能家居环境数据的收集与扩充,并基于数据特点论述采用深度学习方法构建智能家居管控模型的动机;第4节提出基于深度学习的智能家居管控模型 DeepHome,并论述训练数据样本结构与模型训练方法;第5节给出 DeepHome 模型在不同数据集上的实验结果及其分析;最后总结本文工作,提出进一步研究方向。

2 相关工作

智能家居作为一种家居自动化技术早已在学术界和产业界受到广泛关注。当前的众多研究专注于实现设备间的集中交互,诸如 DLNA (Digital Living Network Alliance)^[9]、UPnP^[10]、Z-Wave^[11]

和 Speakeasy^[12]等技术和协议不断提升智能家居、智能硬件设备间的互操作性和通信能力,是智能家居设备的集中管控得以实现的重要技术基础。

在实现设备互联互通的基础上,智能家居管控平台得到了研究者的高度关注。Control4 作为一个强调可扩展性的智能家居自动化系统出现,然而其仅允许系统按照用户预设的操作规则进行管理,难以有效适应家居环境的变化。与 Control4 类似的诸如 HomeSeer、Elk M1、Leviton 等系统也仅能由专业人员进行安装调试,使用比较受限。由微软研究院提出的 HomeOS 则提出了一种类似 PC 操作系统的智能家居管控系统,专注于提高系统可扩展性,降低新硬件、新服务和新功能开发和接入系统的成本,而在系统和设备管理方面则交给开放性平台与外部应用程序^[13]。此外,还有法国运营商 Orange 基于 Z-Wave 技术提出的 Homelive 智能家居解决方案,着眼于家居系统智能化的 eHome 方案^[14],以及麻省理工大学的未来智能家居项目 House_n^[15]等一系列研究课题。

在产业界,智能家居平台同样吸引了众多目光,各大互联网、通信、电子厂商纷纷布局智能家居平台:在国外市场,高通公司于 2013 年组建 AllSeen 联盟,推出 AllJoyn 物联框架;2014 年,谷歌公司旗下的智能家居公司 Nest 推出 Thread 智能框架;同年,苹果公司也推出了智能家居平台 HomeKit。在国内市场,QQ 物联智能硬件平台、京东智能云、BaiduIoT 物联网平台、阿里物联平台等智能硬件平台也相继推出。

清华大学徐恪教授所在课题组提出了面向未来智慧家庭的统一管控平台 SDSH (Software Defined Smart Home),该平台在智能家居环境下引入软件定义网络 (Software Defined Network, SDN) 的策略,使用统一的软件接口实现对智能设备的管控,有效整合了智能家居系统中大量存在的异构网络设备,具备虚拟化、开放化、集中化的特性,能够灵活适应家居环境的多样化差异,满足用户对智能家居的需求^[16]。

本文所提出的 DeepHome 模型,在当前智能家居管控平台和相关系统的基础上,提供智能化的家居设备管理能力,提升现有平台的使用体验并对智能家居领域的进一步发展产生积极影响。

3 动机与挑战

当前的智能家居在集中管理应用中,仍然使用

的是依靠用户人工输入指定规则(例如室温高于 30 摄氏度则打开空调)的方式形成平台的管理逻辑. 这种方式在设备较少的情况下能够取得一定的效果. 然而当设备数量增多, 对智能化要求更高时, 暂且不讨论其操作成本将会在多大程度上影响用户体验, 类似复杂使用场景、多用户需求冲突调解以及突发性事件的发生, 将使得依靠用户提前输入运行规则进行系统控制的方式很快难以满足用户和系统需求. 在这种情况下, 一个具备人工智能的自动化、集中式智能管理算法将是智能家居平台的必备技术以及核心竞争力所在.

与其他机器学习方式相比, 深度学习通过多层次的特征构建而得到对于原始数据的不同抽象层次的表示, 进而提高分类和预测等任务的准确性. 基于复杂家居环境的智能设备状态决策, 可以视为一个多变量非线性的复杂函数. 鉴于深度学习对非线性复杂函数的强大学习能力, 有理由相信其在智能家居的自动化集中控制领域能够取得令人满意的效果. 然而, 在进一步讨论深度学习方法的应用之前, 尚有两方面的挑战和困难需要克服.

3.1 训练数据集的缺乏

深度学习方法非常依赖已有数据集的质量和数

量. 然而, 智能家居行业进入快速发展阶段仅仅是在近三年以内, 受限于硬件产品相对较慢的生产和迭代周期, 智能家居目前仍处于初步发展阶段. 尽管各大互联网巨头均开始以不同角度布局智能家居平台, 但受限于当前智能家居硬件设备的品类、用户安装的数量和硬件本身监测功能, 尚未有一家平台能够积累足够数量或质量的用户数据. 训练数据的缺乏, 是深度学习方法应用于智能家居领域所面对的第一个挑战.

本文所用智能家居环境原始数据的收集与生成过程如图 1 所示. 首先, 以在线问卷形式调研用户居家生活习惯^①. 调查问卷包含 23 个问题, 主要涉及用户作息时间、用户居家行为习惯以及用户年龄、性别、细心程度等用户个人特征. 总计收集到有效问卷 1753 份, 主要包含长期居家人士、在职人士和学生三类人群. 基于上述问卷调研结果, 整理并形成了典型用户的家居生活习惯信息. 与此同时, 设计并部署了一套在线信息提交系统(如图 2 所示), 用户能够根据其实际居住环境与家电信息进行设计布局, 并通过该系统上传家电操作记录^②. 课题组邀请不同生活习惯的志愿者, 通过该系统提交日常生活中对家电的操作记录, 总计收到 3342 条有效信息.

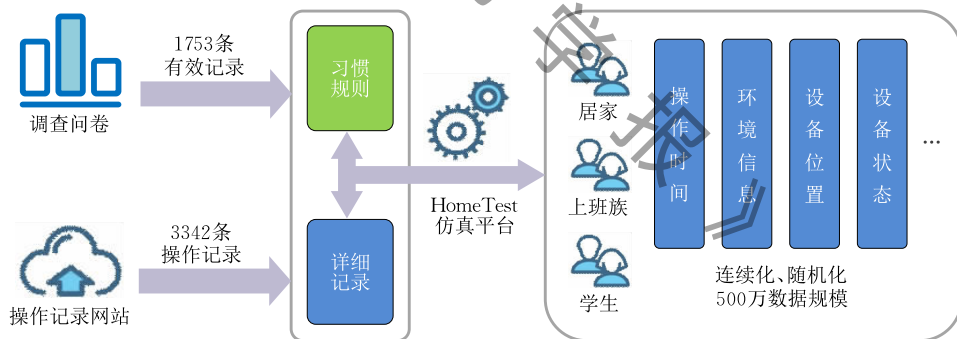


图 1 原始训练集数据的生成

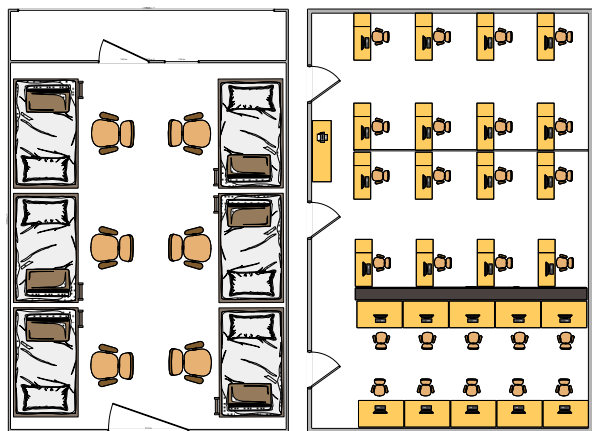


图 2 家电操作上传系统

在上述调研与数据收集工作基础上, 本文构建了一种智能家居模拟测试平台 HomeTest^③. 该平台以用户家庭空间结构、家用电器位置与种类、用户性格特征等要素为输入, 基于用户作息时间表, 依时间序列对家居场景进行模拟. 该平台由事件队列驱动, 每个发生的事件将产生相应的用户、设备、环境状态的变化. 不断变化的家居状态与时间、用户作息时间表结合, 触发新事件加入事件队列. HomeTest 平台基于用户的年龄、性别、细心程度、生活规律性等特

① <https://sojump.com/jq/9653672.aspx>

② <http://121.199.48.193/shdata/index.html>

③ <https://github.com/mb003/HomeTest>

征,为事件的触发以及相关用户、设备状态的改变增加随机参数,增强其对家居环境中随机因素的模拟能力。

在研究过程中,邀请五名典型志愿者向 Home-Test 平台输入其家居环境、个人特征、作息时间。五位志愿者中,包含一位初中学生、一位在读硕士研究生,两位在职人士以及一位退休人士。基于上述输入,HomeTest 平台随机化、连续化地模拟生成每位志愿者一年的家居数据(每 30 秒对家庭环境数据、用户状态、设备状态进行采样),总计约 500 万条。模拟过程以可视化界面呈现给对应志愿者,由志愿者评判模拟的用户行为是否与其自身相近。五名志愿者分别对连续 30 天的模拟行为进行评估,在得到评价的 150 天中,仅有 9 天的模拟行为被志愿者判定为与其生活习惯不符。因此,HomeTest 平台对家居场景的模拟数据是可信的。上述 500 万条纪录将用于后续模型的训练,解决数据集引入问题。

3.2 家居环境的差异性

家庭环境中对家电的使用需求,因不同的环境、用户、气候等因素会产生很大差异,而家居生活中大量的突发事件和用户行为的随机性也会大大增加管控算法的设计难度。

基于人工输入的设备控制规则或是基于浅层神经网络的机器学习方法进行决策,往往存在可移植性与准确率的矛盾:追求在特定家庭中更高的决策准确率,往往需要针对具体环境进行具体调整,此时的模型往往不能在新的环境下保持可以接受的决策准确率,可移植性较低;而具有高可移植性的通用模型,在具体家居环境下准确率较低,需要较长的训练时间或较多的人力调整来逐渐提高决策准确率。

深度学习方法常常在进行整体模型训练之前,使用无监督方法对部分隐层参数进行预训练,以得到训练数据的更为抽象的特征,克服随机梯度下降方法在深层网络训练时遇到的困难^[17]。在智能家居环境中,特征预训练的方法对于缓解可移植性与准确率的矛盾有启发意义:对不同家庭不同类别的设

备进行统一建模,进行多层无监督预训练,以获得设备级的通用抽象特征。在具体家庭进行部署时,用预训练模型对各设备模型进行初始化,再将各设备的抽象特征汇总进行完整模型训练。抽象的设备特征提升了模型的通用性,而完整模型的训练可以充分发掘具体环境下设备间的关联关系,提升在具体环境下的决策准确率。

4 DeepHome 模型设计与训练

本节首先讨论面向智能家居环境的数据集样本结构;然后基于结构化的数据集提出 DeepHome 深度学习模型结构;最后,具体论述 DeepHome 模型中各参数的训练方法。

4.1 数据样本结构

智能家居环境中众多的传感器、智能设备时刻产生大量数据,众多的数据为全面挖掘用户行为,实现家居设备智能化管控提供了基础。

智能家居环境数据通常包括监测数据与智能设备状态。监测数据分为两部分,一部分是包括温度、湿度、时间、天气、亮度等数据在内的环境监测数据,一般通过传感器设备和网络信息获取;另一部分是包括位置、状态、动作等在内的用户信息,通过室内定位技术、可穿戴设备等途径获取。监测数据均为单向流动,智能家居中控系统对该类数据只能采集而不能直接修改、操作。智能设备状态指当前智能设备的工作状态和参数设定,一方面其作为设备信息成为 DeepHome 模型进行决策的部分输入维度,另一方面也是 DeepHome 模型的决策目标,模型基于当前环境数据进行决策,通过修改设备状态实现家居智能管控。

由于智能家居环境数据分布式、多渠道的收集方式,整合零散的数据是进行设备控制决策的重要前提。本文提出一种家居环境数据样本的表述规范:每隔 30 秒对当前家居环境数据进行采样,并整理为如图 3 所示结构的一条数据样本,作为当前家居环境的模式化表征,构成原始数据集。

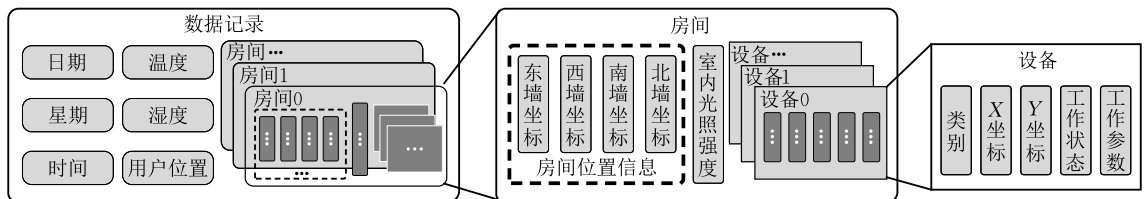


图 3 原始数据集数据格式

智能家居中控系统对家居环境的智能化管控, 是通过智能设备的工作状态和工作参数的控制实现的. 对当前智能设备应处于的工作状态与参数的决策, 一方面基于当前时刻的包括环境数据和用户信息的监测数据; 另一方面, 一段时间之前的监测数据与设备状态往往隐含着对当前决策的相关信息, 且其数据相关性与时间间隔呈负相关. 因此, 为决策某一时刻的设备状态, 其决策样本集 T 的每条样本包括:

当前环境信息: 原始数据集中对应当前时间的数据样本, 但不包含各设备的工作状态项和工作参数项;

历史样本数据: 6 个时间点的数据样本中的温度、湿度、用户位置、室内亮度以及所有设备状态数据, 若当前时间为 T_0 , 则这 6 个时间点为

$$T_i = T_0 - 30 \times 2^{i-1}, i = 1, 2, 3, 4, 5, 6 \quad (1)$$

决策标签: $\mathbf{R} = \{s_1, v_1, \dots, s_n, v_n\}^T$, 其中 s_i 与 v_i 分别表示设备 i 当前工作状态与工作参数.

基于上述样本结构, 在某一时刻, DeepHome 模型以历史数据、家居环境数据、用户信息作为特征输入, 输出对当前时刻各设备状态与参数的预测, 并依照预测对智能家居设备进行控制.

4.2 基于深度学习方法的 DeepHome 模型

在对决策样本和决策目标有明确定义后, 本文

提出一种基于深度学习方法进行智能家居环境决策的模型: DeepHome. DeepHome 模型是一种多隐层深度学习模型, 模型架构如图 4 所示, 包括输入层 L_{in} , 设备级隐层 L_1 与 L_2 , 全连接隐层 L_3 , 以及输出层 L_{out} . L_{in}, L_1, L_2 三层依设备数量 n 分割为 n 个独立域. 在每个域中, 其输入层 L'_{in} 有 56 个神经元, 使用 56 维向量 $\mathbf{L}'_{in} = \{l_1^{in'}, l_2^{in'}, \dots, l_{56}^{in'}\}^T$ 表述, 输入数据从决策样本中截取, 数据结构如图 5 所示, 并将各数据按本身数据范围压缩至区间 $[-1, 1]$.

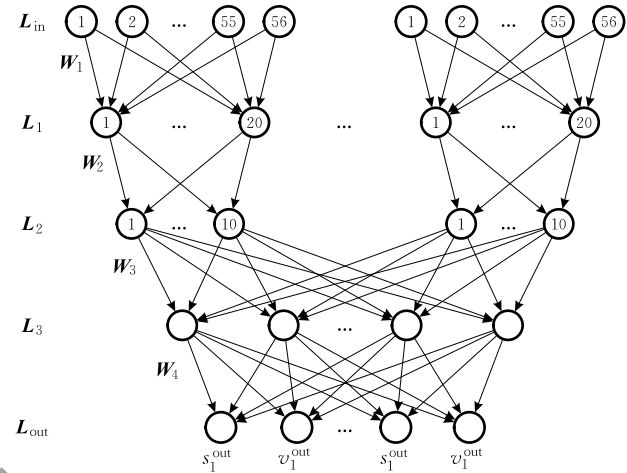


图 4 DeepHome 模型网络结构

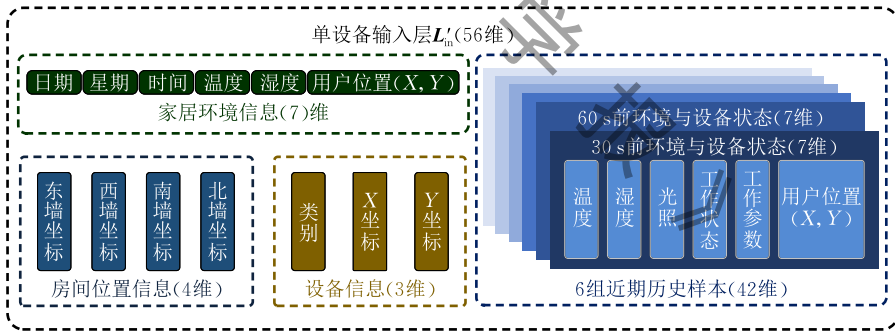


图 5 单设备输入层数据结构

设备级隐层 L'_1 具有 20 个神经元, 各神经元输出以 20 维向量 $\mathbf{L}'_1 = \{l_1^{l'}, l_2^{l'}, \dots, l_{20}^{l'}\}^T$ 表示; 隐层 L'_2 具有 10 个神经元, 神经元输出以 10 维向量 $\mathbf{L}'_2 = \{l_1^{l'}, l_2^{l'}, \dots, l_{10}^{l'}\}^T$ 表示. 域内各层神经元间采用全连接结构. 连接 L'_{in} 与 L'_1 的权重矩阵 \mathbf{W}'_1 定义为

$$\mathbf{W}'_1 = \begin{bmatrix} \omega_{1,1}^1 & \cdots & \omega_{1,56}^1 \\ \vdots & \ddots & \vdots \\ \omega_{20,1}^1 & \cdots & \omega_{20,56}^1 \end{bmatrix} \quad (2)$$

L'_1 层神经元的偏置 \mathbf{B}'_1 为 20 维向量 $\{b_1^{l'}, b_2^{l'}, \dots, b_{20}^{l'}\}^T$, 神经网络节点的激活函数使用双曲正切函数:

$$\tanh(x) = \frac{e^x - e^{-x}}{e^x + e^{-x}} \quad (3)$$

则 L'_1 层第 i 个神经元节点的输出为

$$l_i^{l'} = \tanh\left(\sum_{j=1}^{56} (\omega_{i,j}^1 \times l_j^{in'}) + b_i^{l'}\right) \quad (4)$$

为方便表述, 下文统一使用矩阵运算表示, 则式(4)表述为

$$\mathbf{L}'_1 = \tanh(\mathbf{W}'_1 \cdot \mathbf{L}'_{in} + \mathbf{B}'_1) \quad (5)$$

同样, 本文定义连接 L'_1 层与 L'_2 层的权值矩阵 \mathbf{W}'_2 以及 L'_2 层神经元的偏置向量 \mathbf{B}'_2 :

$$\mathbf{W}'_2 = \begin{bmatrix} \omega_{1,1}^2 & \cdots & \omega_{1,20}^2 \\ \vdots & \ddots & \vdots \\ \omega_{10,1}^2 & \cdots & \omega_{10,20}^2 \end{bmatrix} \quad (6)$$

$$\mathbf{B}'_2 = \{b'_{1,1}, b'_{2,2}, \dots, b'_{10,10}\}^T \quad (7)$$

则 L'_2 层神经元输出由下式求得:

$$\mathbf{L}'_2 = \tanh(\mathbf{W}'_2 \cdot \mathbf{L}'_1 + \mathbf{B}'_2) \quad (8)$$

全连接隐层 L_3 具有 1000 个神经元, 该层神经元输出表示为 $\mathbf{L}_3 = \{l_1^3, l_2^3, \dots, l_{1000}^3\}^T$. 现令 L_2 为各域内 L'_2 的结合, 并与 L_3 层进行全连接, 则有权重矩阵 \mathbf{W}_3 , 偏置向量 \mathbf{B}_3 以及 L_3 层的激活输出值:

$$\mathbf{W}_3 = \begin{pmatrix} \tau_{1,1}^3 & \dots & \tau_{1,10n}^3 \\ \vdots & \ddots & \vdots \\ \tau_{1000,1}^3 & \dots & \tau_{1000,10n}^3 \end{pmatrix} \quad (9)$$

$$\mathbf{B}_3 = \{b_1^3, b_2^3, \dots, b_{1000}^3\}^T \quad (10)$$

$$\mathbf{L}_3 = \tanh(\mathbf{W}_3 \cdot \mathbf{L}_2 + \mathbf{B}_3) \quad (11)$$

最后, 输出层 L_{out} 神经元数量为设备数量 n 的 2 倍, 其输出值即模型决策结果 $\mathbf{L}_{out} = \{s_1^{out}, v_1^{out}, \dots, s_n^{out}, v_n^{out}\}^T$, 依次描述了各设备工作状态、工作参数的决策值. 定义 L_3 层与 L_{out} 层全连接权重矩阵为 \mathbf{W}_4 , 偏置向量为 \mathbf{B}_4 , 则有

$$\mathbf{L}_{out} = \tanh(\mathbf{W}_4 \cdot \mathbf{L}_3 + \mathbf{B}_4) \quad (12)$$

DeepHome 依照 \mathbf{L}_{out} 各项数值对相应设备进行控制.

4.3 DeepHome 模型训练过程

本节论述 DeepHome 模型的训练过程, 包括设备级隐层权重的无监督预训练过程, 以及完整模型的有监督多层前馈神经网络训练过程.

智能家居环境下, 决策样本数据当中包含众多监测数据、设备信息以及历史数据, 其数据维度较高. 采用基于多隐层的前馈神经网络, 模型中存在大量待训练的权重参数, 对于隐层较多的网络而言, 基于梯度下降的反向传播训练方法对于接近输入层的各参数的调整幅度过小, 出现“梯度扩散”现象, 导致模型训练的收敛速度下降^[18]. 另一方面, 过多的参数很容易产生训练样本“过拟合”, 即在训练集决策误差不断下降的同时, 测试集的误差不断增长, 模型泛化能力下降.

为克服深度神经网络的梯度扩散问题, 提升模型泛化能力, DeepHome 针对智能家居场景的数据特点, 采用自编码网络 (Autoencoder) 进行单个设备模型的无监督预训练, 旨在学习对单一设备决策样本的一种压缩、分布式的数据特征^[19]. DeepHome 模型对 L'_1, L'_2 两级隐层的 4 个参数矩阵 $\mathbf{W}'_1, \mathbf{B}'_1, \mathbf{W}'_2, \mathbf{B}'_2$ 进行预训练, 并将预训练所得参数作为完整模型的对应参数的初始化值.

隐层参数矩阵 $\mathbf{W}'_1, \mathbf{B}'_1$ 的 Autoencoder 预训练网

络结构如图 6 所示, 为单隐层全连接前馈神经网络, 分为输入层 $L_{pin} = \{x_1, x_2, \dots, x_{56}\}^T$, 隐层 $L'_1 = \{l'_1, l'_2, \dots, l'_{20}\}^T$ 与输出层 $L_{pout} = \{y_1, y_2, \dots, y_{56}\}^T$. DeepHome 使用随机梯度下降算法进行训练, 并在训练过程中采用 L2 正则化^[20]、弃权 (Dropout)^[21] 两种方法来降低网络过拟合的可能性, 增强预训练的抽象特征对输入数据的表达能力. 训练的单次迭代过程如过程 I 所示.

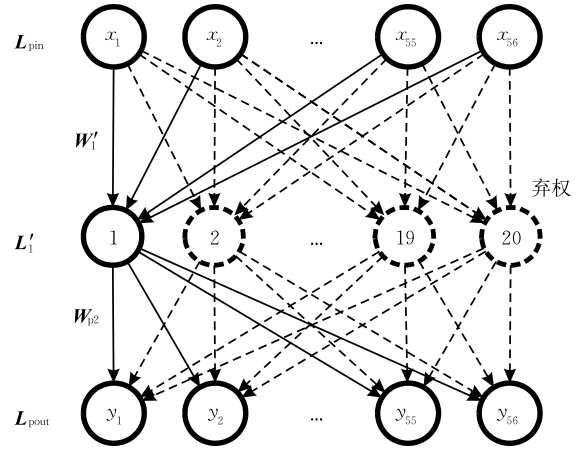


图 6 L'_1 隐层参数训练网络

过程 I. 隐层 L'_1 参数 $\mathbf{W}'_1, \mathbf{B}'_1$ 预训练.

1. 将原始训练集中所有样本依照图 5 格式进行预处理, 得到表示单一设备信息的 56 维样本集 S .
2. 使用高斯分布初始化参数矩阵 $\mathbf{W}'_1, \mathbf{W}_{p2}$: 对 $\forall w \in \mathbf{W}'_1$, 有 $w \sim N(0, 1/57)$; 对 $\forall w \in \mathbf{W}_{p2}$, 有 $w \sim N(0, 1/21)$.
3. 初始化参数矩阵 $\mathbf{B}'_1, \mathbf{B}_{p2}$: 对 $\forall b \in \mathbf{B}'_1$, 有 $b = 0$; 对 $\forall b \in \mathbf{B}_{p2}$, 有 $b = 0$.
4. 随机不放回抽取 S 中 100 条样本组成一个批次的的数据, 作为 \mathbf{B}_{pin} 层的输出值.
5. 计算隐层 L'_1 输出: $\mathbf{L}'_1 = \tanh(\mathbf{W}'_1 \cdot \mathbf{L}'_{pin} + \mathbf{B}'_1)$.
6. 随机选取 10 个 L'_1 层神经元, 置其输出为 0; 另 10 个神经元输出值翻倍 (Dropout 过程).
7. 依公式 $\mathbf{L}_{pout} = \tanh(\mathbf{W}_{p2} \cdot \mathbf{L}'_1 + \mathbf{B}_{p2})$ 计算 \mathbf{L}_{pout} .
8. 计算代价函数:

$$C_1 = \sum_{i=1}^{56} (y_i - x_i)^2 + \mu \sum_{w \in \mathbf{W}'_1, \mathbf{W}_{p2}} w^2,$$

并沿 C_1 梯度下降方向反向调节 $\mathbf{W}_{p2}, \mathbf{B}_{p2}, \mathbf{W}'_1, \mathbf{B}'_1$.

9. 若 S 中样本数量小于 100, 本轮迭代结束; 否则跳转步 4.

经过程 I 的多轮迭代后, 得到参数矩阵 $\mathbf{W}'_1, \mathbf{B}'_1$ 的预训练值, 将其作为常数代入图 7 所示的 Autoencoder 预训练网络, 按过程 II 对隐层 L'_2 的参数 $\mathbf{W}'_2, \mathbf{B}'_2$ 进行预训练. 其中, L_{pin}, L'_1 的定义与过程 I

相同,输出层 $L_{\text{pout}} = \{y_1, y_2, \dots, y_{20}\}^T$.

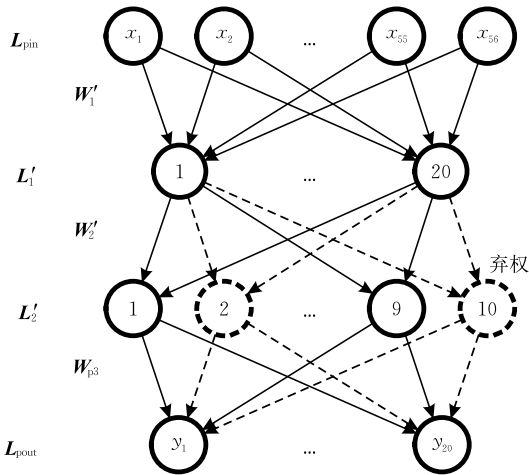


图 7 L'_2 隐层参数训练网络

过程 II. 隐层 L'_2 参数 W'_2, B'_2 预训练.

1. 将原始训练集中所有样本依照图 5 的数据格式进行预处理,得到表示单一设备信息的 56 维样本集 S .
2. 使用高斯分布初始化参数矩阵 W'_2, B'_2 : 对 $\forall w \in W'_2$, 有 $w \sim N(0, 1/21)$, 对 $\forall w \in W_{p3}$, 有 $w \sim N(0, 1/11)$.
3. 初始化参数矩阵 B'_2, B_{p3} : 对 $\forall b \in B'_2$, 有 $b = 0$; 对 $\forall b \in B_{p3}$, 有 $b = 0$.
4. 随机不放回抽取 S 中 100 条样本组成一个批次的的数据,作为 L_{pin} 层的输出值.
5. 计算隐层 L'_1 输出: $L'_1 = \tanh(W'_1 \cdot L_{\text{pin}} + B'_1)$.
6. 计算隐层 L'_2 输出: $L'_2 = \tanh(W'_2 \cdot L'_1 + B'_2)$.
7. 随机选取 5 个 L'_2 层神经元,置其输出为 0;另 5 个神经元输出值翻倍(Dropout 过程).
8. 计算 L_{pout} 输出值: $L_{\text{pout}} = \tanh(W_{p3} \cdot L'_2 + B_{p3})$.
9. 计算代价函数:

$$C_2 = \sum_{i=1}^{20} (y_i - l'_i)^2 + \mu \sum_{w \in W'_2, W_{p3}} \omega^2,$$

并沿 C_2 梯度下降方向反向调节 $W_{p3}, B_{p3}, W'_2, B'_2$.

10. 若 S 中样本数量小于 100,本次迭代结束;否则跳转步 4.

经过程 I 与过程 II 的多轮迭代预训练后,获得单一设备的 Autoencoder 模型. 利用该模型对 DeepHome 模型的设备级隐层相应参数做初始化,继而进行完整模型的训练. DeepHome 完整模型训练单次迭代过程如过程 III 所述.

过程 III. DeepHome 模型训练.

参数: 模型中智能设备数量为 n .

1. 使用预训练所得参数 W'_1, B'_1, W'_2, B'_2 对各域中对应参数初始化.
2. 使用高斯分布初始化参数矩阵 W_3, W_4 : 对 $\forall w \in W_3$, 有 $w \sim N(0, 1/(10n + 1))$; 对 $\forall w \in W_4$, 有 $w \sim N(0, 1/1001)$.

3. 初始化参数矩阵 B_3, B_4 : 对 $\forall b \in B_3$, 有 $b = 0$; 对 $\forall b \in B_4$, 有 $b = 0$.

4. 随机不放回抽取原始决策样本集 T 中 100 条样本组成一个批次的的数据,按设备依次整理为图 5 所示结构,并作为各设备域中 L'_m 层的值.

5. 依据式 (5)、(8)、(11)、(12) 计算 L_{out} .

6. 计算代价函数:

$$C = \sum_{i=1}^n (l_{2i}^{\text{out}} - s_{2i-1})^2 + \sum_{i=1}^n (l_{2i}^{\text{out}} - v_{2i})^2 + \mu \sum_{w \in W_3, W_4} \omega^2,$$

并沿 C 梯度下降方向反向调节所有权重矩阵和偏置向量.

7. 若 T 中样本数量小于 100,本次迭代结束;否则跳转步 4.

在实际应用中,过程 I、II 面向模型通用化能力,基于不同用户的智能家居数据进行训练. 过程 III 面向特定用户环境,在用户家中进行个性化训练: 首先,基于具体设备数量构建 DeepHome 模型,使用过程 I、II 所得通用设备模型对模型设备域参数进行初始化. 然后,依本文 4.1 节所述方式,基于智能家居平台持续对家居环境数据采样. 经模型部署的最初一个月的数据收集后,依过程 III 进行离线训练,获得初始模型. 之后一个月,智能家居平台将模型做出的改变设备状态的预测指令发送给用户,并依照用户对指令的同意或反对意见,对模型进行在线训练. 在部署两个月后,平台开始依据模型决策结果对家居设备进行自动控制,同时持续进行数据采样用于在线训练,以适应家居环境可能的变化.

DeepHome 模型经过程 I、II、III 的训练后,具备了基于家居环境数据进行智能化管控的能力.

5 实验分析

5.1 实验设计

为研究 DeepHome 模型的性能,本文设计并进行了 4 组相关实验,分别测试模型的准确性、模型的收敛速度、模型参数对算法结果的影响,以及模型对真实数据样本的预测能力.

实验数据的获取方式如本文第 3 节所述,主要分为预训练集、训练集和测试集. 首先,将本文 3.1 节所述基于 HomeTest 平台的 500 万条仿真数据,拆分为如本文第 4 节所述单设备域样本与标签,进行通用单设备模型预训练. 然后,使用 HomeTest 平台模拟一位学生 2015 年全年的日常生活,得到总计约 110 万条记录(每 30 秒一条记录),依本文 4.1 节所述数据样本结构整理为训练集.

考虑到智能家居环境中家电设备的工作状态发

生改变的時刻仅占约 0.53%,为了更加全面的考察 DeepHome 模型的实际性能,测试集主要分为两部分:测试集 A 和测试集 B.其中,完整的测试集数据,由 HomeTest 平台模拟与训练集相同的用户在 2016 年的日常生活数据.测试集 A 在完整数据的集合中进行抽样,获取总计约 10 万条记录,其中将包含大量家电设备工作状态未发生变化的样本.测试集 B 则只抓取完整数据集当中,设备状态发生改变的時刻的数据记录,数据量总计为 5864 条.如本文 4.1 节所示,测试集 A、B 各样本的标签为该時刻各设备的工作状态与工作参数.

模型训练与实验基于 TensorFlow 框架进行. TensorFlow 是基于数据流图计算的开源深度学习框架,其灵活的架构使得深度学习方法能够在桌面、服务器或移动设备的单个或多个 CPU 和 GPU 上进行部署与计算^[22]. TensorFlow 最早由 Google Brain 团队的研究人员和工程师研发,目的是管理机器学习和深度神经网络的研究工作,但该系统的高通用性使其能够适用于众多应用领域. TensorFlow 自 2015 年底开源,并于近期开源其分布式版本,受到广泛关注.

5.2 决策准确度

本实验对 DeepHome 模型的预测准确度进行分析. DeepHome 模型针对每一个样本输入,将给出对每个设备的运行状态(开/关)、工作参数(如空调温度)的预测.对于运行状态,将 DeepHome 的输出与提前设定的阈值进行对比,小于阈值的情况判断输出为 0(关),其它情况判断输出为 1(开).之后,将 DeepHome 的输出与测试集的结果标签中相应设备的运行状态进行对比,完全一致则说明运行状态判断正确.与此同时,求 DeepHome 预测的工作参数与标签中对应的正确工作参数的差值的平方和,若平方和小于 0.05 则认为 DeepHome 进行了一次正确的预测.

作为对比,一方面,选用双隐层前馈神经网络模型,直接使用随机梯度下降算法训练,不进行逐层预训练;另一方面,选用基于人工逻辑规则的方法进行智能设备的逐一设定.其中,双隐层神经网络采用全连接结构,与输入层全连接的隐层具有 2000 个神经元,与输出层全连接的隐层具有 1000 个神经元,激活函数均使用 tanh;人工逻辑规则基于用户作息时间表以及环境状态、用户状态进行设备状态判定,占比最大的逻辑规则包括两类:设备定时开关控制、基于用户所在房间进行设备开关控制.

本实验在测试集 A 与测试集 B 上使用三种方案进行决策,并根据上述准确度判断准则,计算不同方案在不同迭代轮数下的决策准确率.实验结果如图 8 和图 9 所示.

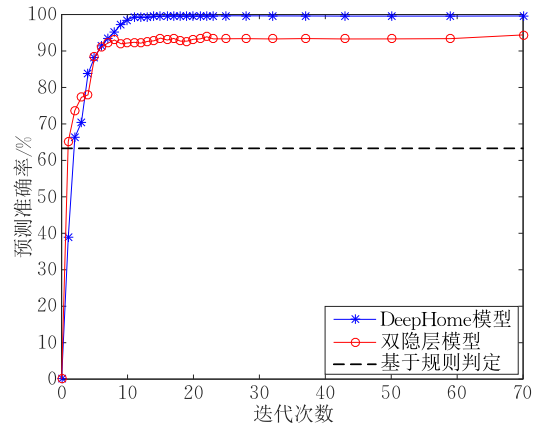


图 8 不同模型在测试集 A 上的预测准确率

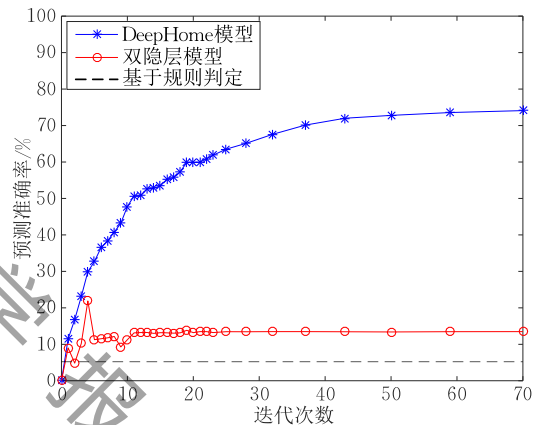


图 9 不同模型在测试集 B 上的预测准确率

通过实验结果可知,两种机器学习算法随着训练轮次的增加,判断准确率均有明显提升,而逻辑规则不随迭代次数的增加而改变,其准确率为定值.在测试集 A 上,DeepHome 通过约 10 轮迭代训练,决策准确率得到快速提升,达到约 99.38%.之后通过额外的 60 轮训练,其决策准确率始终维持在 99.6% 上下.双隐层神经网络模型在测试集 A 上通过 5 轮训练即可快速接近最优值,达到约 88.46% 的决策准确率.再通过额外 15 轮训练,即可达到最优算法准确度,约 93.03%.与 DeepHome 相比,最优准确度低约 6%.而基于逻辑规则的方案,其决策准确率仅为 63.3%,远低于两种机器学习模型.

在测试集 B 上,由于去除了大量重复数据,仅关注所有前后状态发生变化的记录,3 种方案的准确率相比测试集 A 均产生了一定程度的下降.其中,DeepHome 模型经过 70 轮训练,最优可达到约

74.09%的准确率,相比测试集 A,准确率下降约 25%。对于双隐层神经网络模型,准确率稳定于 13%左右,下降约 80%。基于逻辑规则的方案,准确率则只有 5.2%。

事实上,对于测试集 A 的预测准确率代表了模型能够使家居设备处于正确工作状态的时间占比;对于测试集 B 的预测准确率代表了在用户出现设备状态改变需求时,模型及时预判并作出正确响应的准确率。比较不同方案的准确率可以发现,DeepHome 在测试集 A 和测试集 B 上均明显优于其他方案。特别是对于测试集 B,DeepHome 模型仍然达到可以接受的准确率 74%,而双隐层神经网络模型的准确率仅能达到 13%左右,存在根本性差距。通过该实验可知,DeepHome 模型的准确率完全可以满足实际使用的需求,相对于基于逻辑规则判断的方案和浅层神经网络模型,其准确度有明显提升。

5.3 训练收敛速度

DeepHome 模型作为一种机器学习技术,其模型的有效性在很大程度上将受限于训练数据的数量,其收敛速度将是基础性能的一个重要指标。

本项实验基于本文第 4 节所描述的误差函数随训练样本数量的变化趋势,分析 DeepHome 模型训练的收敛速度,结果如图 10 所示。

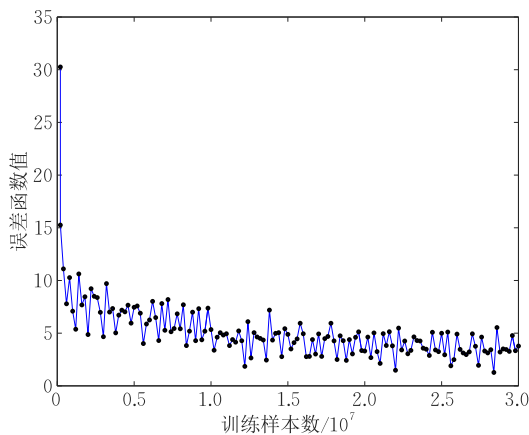


图 10 误差函数随训练样本数量的变化趋势

基于上述实验结果可知,DeepHome 模型在训练过程中,其单批次(100 条样本)训练数据误差函数总值呈不断收敛趋势,由最初的 30.27 收敛至最低 1.28 左右,此时训练样本数据量增长至 3000 万(约 30 轮迭代)。在训练数据量增长到 500 万(约 5 轮迭代)时,误差函数总值已经收敛至约 4.02。由此可以看出,在经过约 5 轮迭代后,DeepHome 算法的收敛速度明显放缓。在使用单 GPU 加速的训练环境下,每一轮迭代所需时间约为 12.4 分钟。

在实际应用时,家居数据不断更新,模型也需要在数据集有较大更新时进行离线训练。考虑到承担模型训练的智能家居中控设备,其处理器性能往往较低,权衡训练速度与准确率等实用性要求,过高的迭代次数没有意义,每次训练进行约 5 至 10 轮迭代是合理的方案。

5.4 模型参数

5.4.1 设备状态阈值对准确率的影响

如本文 5.2 节所述,DeepHome 模型在对输出进行判断时,通过与预先设定的阈值进行比较,当输出结果小于阈值时,最终输出设定为 0,否则设定为 1。由此可以看出,如何设定阈值将在一定程度上影响 DeepHome 模型的准确率。为此,设计如下实验研究阈值的选取:选取本文 4.2 节实验中经过 70 轮迭代所得 DeepHome 模型,在不同阈值设定下使用测试集 A、B 进行测试。其中,阈值以 0 和 1 为间隔在区间 $[0,1]$ 取值,实验结果如图 11 所示。

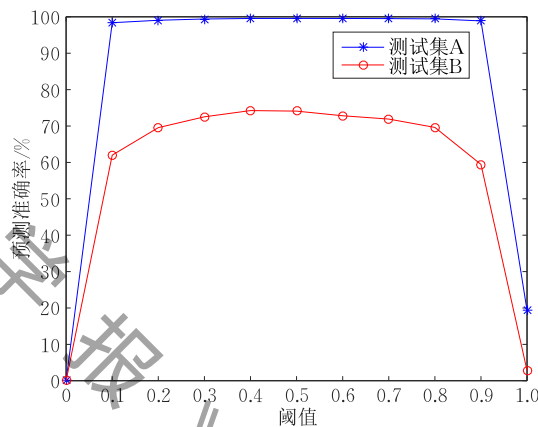


图 11 阈值对准确率的影响

通过实验结果可以发现,在测试集 A 上,DeepHome 模型的阈值在区间 $[0.1,0.9]$ 内改变,对预测准确率的影响并不明显。最低的准确率出现在阈值为 0.1 时,约为 98.39%,而最大精度为 99.58%,由阈值为 0.5 时取得,极值之间相差仅为 1%左右。而在测试集 B 上,同样的实验准确率差异较大。其中准确率最低为 59.45%,在阈值为 0.9 时取得,而最大准确率为 74.25%当阈值为 0.4 时取得,极值之间相差约 15%。

一方面,阈值对准确率的影响较低,说明 DeepHome 模型对各设备状态的预测值基本集中在 0、1 两端,这反映出 DeepHome 模型只在极少数情况下不能给出设备开、关的肯定性预测。另一方面,通过过滤掉测试集 A 中状态重复数据的干扰,在测试集 B 上,靠近两端的阈值选择会带来准确率的显著降

低,因此,在大多数情况下,使用 0.5 作为阈值是一个可行的选择,而不同训练集自身的差异将造成阈值在 0.5 周围的小幅度偏移,最优选择还需针对具体数据集调整。

5.4.2 设备数量对准确度的影响

当智能家庭环境中智能设备的数量达到一定程度后,无论是每一条描述当前家居整体状态的数据,还是单个设备的关联信息的数据,其维度都会显著增加。在同一时刻正确预测多个设备状态与参数的难度也远远大于对单一设备的预测。为了对这一点进行进一步研究,分别使用 DeepHome 模型、双隐层神经网络模型以及基于逻辑规则的方案,在测试集 A 上进行相关实验,结果如图 12 所示。

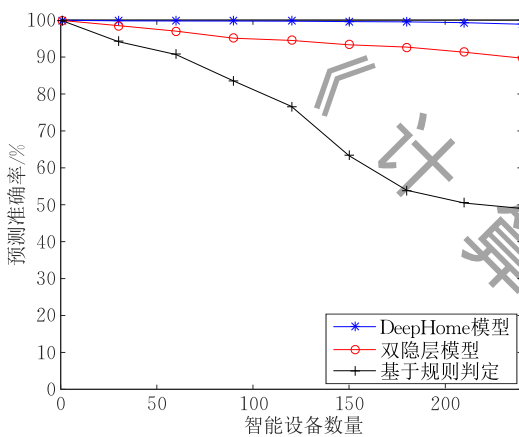


图 12 设备数量对预测准确率的影响

由图中数据可以看出,随着设备数量的增加,三种方案的准确度均呈现出一定程度的下降趋势。其中,DeepHome 模型的下降幅度较小,由 30 个设备时约 99.8% 的预测准确率,下降到 240 个设备时,约 98.9% 的准确率,降幅约为 0.9%。而双隐层神经网络模型,则从最高 93% 的准确率,下降到约 89.7%,降幅约为 3.5%。可以发现,当家庭环境中,智能设备的数量增长时,预测难度增加,三种方案的预测准确度都发生了不同程度的降低,而这其中 DeepHome 模型降幅非常微弱,在智能设备数量增加到非常大的 240 个时(可预见的未来中几乎不会超过这一数量),DeepHome 模型的准确度依旧非常出色,降幅仅为 0.9%。

5.5 真实家居场景数据集实验

为验证 DeepHome 方案在真实家居环境中的适应能力,本文基于在线提交系统收集的用户实际数据测试 DeepHome 的有效性。

如第 3 节所述,已经通过在线信息提交系统收集了 3342 条不同用户的家居环境及设备状态数据。

其中提交数量最多的用户(下称用户 S1)提交了 1072 条数据,记录的设备数量为 12 个,记录时间持续 83 天;提交数目次多的用户(下称 S2)提交了 663 条数据,记录的设备数量为 17 个,记录时间持续 53 天。由于用户通常只在开关某个设备时进行提交,可以认为用户在两次提交之间设备状态保持不变。因此,对于用户 S1、S2,依照 DeepHome 模型需要的数据集格式进行数据整理:自用户首条提交时间开始,每 30 秒进行一次采样记录,其设备状态与采样时刻 t 之前最近一次提交相同,环境数据依照 t 前后两次提交的值进行加权平均,权值为提交时刻与 t 的时间差。

经数据整理后,对用户 S1、S2 分别随机取 80% 的数据构成训练集 T1、T2,余下 20% 的数据分别构成测试集 A1、A2。为充分评估模型对于用户需求的响应速度,分别截取 T1、T2 中与上一时刻设备状态不同的记录,构成测试集 B1、B2。通常,用户在家中改变设备状态,是为了满足其自身需求。因此,当模型对设备状态的预测与下一时刻用户对设备的操作结果(或者什么也不做,保持设备状态不变)相吻合,即可认为 DeepHome 的该次决策是满足用户需求的。测试所得的预测准确率,一定程度上代表了用户对 DeepHome 管控能力的满意程度。

使用训练集 T1 对 DeepHome 模型进行 50 轮迭代训练,并使用 A1、B1 进行测试;同样的,使用训练集 T2 进行 50 轮迭代训练,使用 A2、B2 进行测试。各测试集预测准确率如图 13 所示。

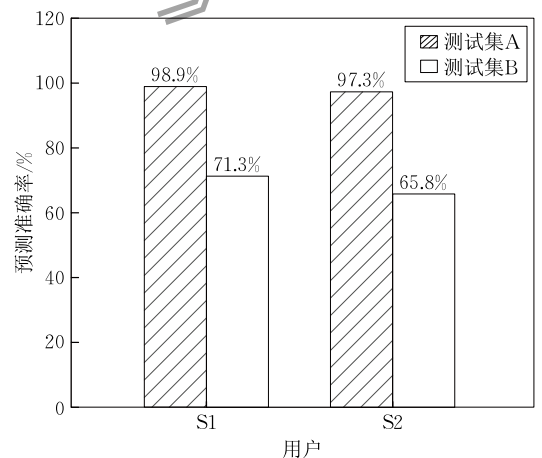


图 13 真实数据集下 DeepHome 模型预测准确率

由实验结果可知,对于用户 S1、S2,DeepHome 模型的决策在 98.9%、97.3% 的时间是符合用户需求的,而对于用户对设备状态进行改变的预测准确率达到 71.3%、65.8%。一方面,用户 S1 的提交数

据较用户 S2 丰富,设备数量却更少,因此对用户 S1 的预测更为准确. 另一方面,面对实际家居环境数据,DeepHome 模型在约 80 天的记录数据训练之下,其预测准确率达到可以接受的 98.9%,证明了该模型在实际环境中能够为用户提供可信的智能家居自动控制服务.

6 结束语

在智能家居场景下,与传统的基于人工逻辑控制的方案,以及基于浅层神经网络的模型相比,DeepHome 模型针对智能家居环境数据特点,基于深度学习方法进行模型构建. 一方面,通过 Autoencoder 方法进行预训练,在缓解深度神经网络训练过程中的梯度扩散效应的同时,使模型获得了通用性设备数据特征,提升了模型的泛化能力. 另一方面,DeepHome 模型的多隐层结构能够充分拟合复杂的多设备协同环境. 在实验环境中,DeepHome 在 99.6% 的时间内能够使各智能设备状态符合用户需求,在约 74% 的情况下能够准确预测用户行为并调整相应设备状态. 因此,本文提出的 DeepHome 模型可以作为一种有效的家居智能化管控方案.

随着智能家居产业的生长,智能家居设备保有量不断上升,智能家居平台逐步推广,有价值的智能家居环境数据将不断增多. 在未来研究中,一方面需要不断拓展数据来源渠道,使用更多、更有效的数据训练 DeepHome 模型;另一方面需要尝试将 DeepHome 模型与不断成熟的商业智能家居平台相结合,以期得到性能更好,更适用于智能家居实际环境的自动管控模型.

参 考 文 献

- [1] Brush A J, Lee B, Mahajan R, et al. Home automation in the wild: Challenges and opportunities//Proceedings of the SIGCHI Conference on Human Factors in Computing Systems. Vancouver, Canada, 2011: 2115-2124
- [2] Edwards W K, Grinter R E, Mahajan R, et al. Advancing the state of home networking. *Communications of the ACM*, 2011, 54(6): 62-71
- [3] Grinter R E, Edwards W K, Chetty M, et al. The ins and outs of home networking: The case for useful and usable domestic networking. *ACM Transactions on Computer-Human Interaction*, 2009, 16(2): 8
- [4] Papadopoulos N, Meliones A, Economou D, et al. A connected home platform and development framework for smart home control applications//Proceedings of the 7th IEEE International Conference on Industrial Informatics. Cardiff, United Kingdom, 2009: 402-409
- [5] Michalski R S, Carbonell J G, Mitchell T M, eds. *Machine Learning: An Artificial Intelligence Approach*. Berlin Heidelberg, Germany: Springer Science & Business Media, 2013.
- [6] Sebastiani F. Machine learning in automated text categorization. *ACM Computing Surveys*, 2002, 34(1): 1-47
- [7] Schmidhuber J. Deep learning in neural networks: An overview. *Neural Networks*, 2015, 61: 85-117
- [8] LeCun Y, Bengio Y, Hinton G. Deep learning. *Nature*, 2015, 521(7553): 436-444
- [9] Kawamoto R, Emori T, Sakata S, et al. DLNA-ZigBee gateway architecture and energy efficient sensor control for home networks//Proceedings of the 2007 16th IST Mobile and Wireless Communications Summit. Budapest, Hungary, 2007: 1-5
- [10] Kim D S, Lee J M, Kwon W H, et al. Design and implementation of home network systems using UPnP middleware for networked appliances. *IEEE Transactions on Consumer Electronics*, 2002, 48(4): 963-972
- [11] Gomez C, Paradells J. Wireless home automation networks: A survey of architectures and technologies. *IEEE Communications Magazine*, 2010, 48(6): 92-101
- [12] Edwards W K, Newman M W, Sedivy J Z, et al. Using speakeasy for ad hoc peer-to-peer collaboration//Proceedings of the 2002 ACM Conference on Computer Supported Cooperative Work. Chicago, USA, 2002: 256-265
- [13] Dixon C, Mahajan R, Agarwal S, et al. An operating system for the home//Proceedings of the 9th USENIX Symposium on Networked Systems Design and Implementation (NSDI 12). San Jose, USA, 2012: 337-352
- [14] Bregman D. Smart home intelligence—The eHome that learns. *International Journal of Smart Home*, 2010, 4(4): 35-46
- [15] Intille S S. Designing a home of the future. *IEEE Pervasive Computing*, 2002, 1(2): 76-82
- [16] Xu K, Wang X, Wei W, et al. Toward software defined smart home. *IEEE Communications Magazine*, 2016, 54(5): 116-122
- [17] Glorot X, Bengio Y. Understanding the difficulty of training deep feedforward neural networks//Proceedings of the 13th International Conference on Artificial Intelligence and Statistics. Sardinia, Italy, 2010: 249-256
- [18] Bengio Y, Simard P, Frasconi P. Learning long-term dependencies with gradient is difficult. *IEEE Transactions on Neural Networks*, 1994, 5(2): 157-166
- [19] Bengio Y, Lamblin P, Popovici D, et al. Greedy layer-wise training of deep networks. *Advances in Neural Information Processing Systems*, 2007, 19: 153

- [20] Cortes C, Mohri M, Rostamizadeh A. L2 regularization for learning kernels//Proceedings of the 25th Conference on Uncertainty in Artificial Intelligence. Montreal, Canada, 2009; 109-116
- [21] Srivastava N, Hinton G E, Krizhevsky A, et al. Dropout: A simple way to prevent neural networks from overfitting. Journal

of Machine Learning Research, 2014, 15(1): 1929-1958

- [22] Abadi M, Barham P, Chen J, et al. TensorFlow: A systems for large-scale machine learning//Proceedings of the 12th USENIX Symposium on Operating System Design and Implementation (OSDI). Savannah Georgia, USA, 2016



MAO Bo, born in 1991, M. S. His research interests include wireless networks, wireless sensor networks application and machine learning.

XU Ke, born in 1974, Ph.D., professor, Ph.D. supervisor. His main research interests include Internet architecture.

high performance router, P2P network, Internet of Things and network economics.

JIN Yue-Hui, born in 1965, professor. Her main research interests include Internet performance measurement and user experience, social networking and recommendation system.

WANG Xiao-Liang, born in 1986, Ph.D. candidate. His research interests include wireless network and mobile computing.

Background

How to control smart home devices intelligently is one of core problems in the field of smart home. The most common way to control the smart devices is based on the manual logic rules, which cannot deal with the unplanned events, cannot deal with the uncertainty of smart home environment and always be complicated to be set. Based on the deep neural network's strong learning abilities of complex problems, this paper presents DeepHome, a deep learning model for controlling smart home devices. Our experiments show that DeepHome can adapt the smart home environment effectively, predict the working status of smart devices accurately and adjust the status by predictions.

Our group committed ourselves to the study of smart home's platformization and the relevant supporting technologies. In the field of smart home architecture, we proposed software defined smart home platform, based on the principles of software defined networking. The design features of virtualization, openness and centralization can effectively integrate the heterogeneous network devices in the smart home

platform, and flexibly adapt to the great difference between family scenes and user demands. In the field of supporting technologies, based on the protocol of Bluetooth Low Energy 4.1 (BLE 4.1), we presented a Bluetooth scatternet formation technology focused on energy efficiency to link the increasing smart home devices. As a method of requirements acquirement in smart home, we designed a high-available and energy-efficient algorithm for indoor location, focused on wearable smart devices.

This work was supported by the National Natural Foundation of China (61170292, 61472212), the National Science and Technology Major Project of China (2015ZX03003004), the National Basic Research Program (973 Program) of China (2012CB315803), the National High Technology Research and Development Program (863 Program) of China (2013AA013302, 2015AA015601), the EU Marie Curie Actions CROWN (FP7-PEOPLE-2013-IRSES-610524), and the Multidisciplinary Fund of Tsinghua National Laboratory for Information Science and Technology.